Parametric Proportional Hazard Models with Applications in Survival analysis

A Thesis Submitted to the

College of Graduate and Postdoctoral Studies

in Partial Fulfillment of the Requirements

for the degree of Doctor of Philosophy

in the Department of Mathematics and Statistics

University of Saskatchewan

Saskatoon

By

Saima Khan Khosa

Permission to Use

In presenting this thesis in partial fulfilment of the requirements for a Postgraduate degree from the University of Saskatchewan, I agree that the Libraries of this University may make it freely available for inspection. I further agree that permission for copying of this thesis in any manner, in whole or in part, for scholarly purposes may be granted by the professor or professors who supervised my thesis work or, in their absence, by the Head of the Department or the Dean of the College in which my thesis work was done. It is understood that any copying or publication or use of this thesis or parts thereof for financial gain shall not be allowed without my written permission. It is also understood that due recognition shall be given to me and to the University of Saskatchewan in any scholarly use which may be made of any material in my thesis.

Requests for permission to copy or to make other use of material in this thesis in whole or part should be addressed to:

Head of the Department of Mathematics and Statistics

Room 142 McLean Hall

106 Wiggins Road

University of Saskatchewan

Saskatoon, Saskatchewan S7N 5E6

Canada

OR

College of Graduate and Postdoctoral Studies

University of Saskatchewan

116 Thorvaldson Building, 110 Science Place

Saskatoon, Saskatchewan S7N 5C9 Canada.

Abstract

Proportional hazard (PH) models can be formulated with or without assuming a probability distribution for survival times. The former assumption leads to parametric models, whereas the latter leads to the semi-parametric Cox model which is by far the most popular in survival analysis. However, a parametric model may lead to more efficient estimates than the Cox's model under certain conditions. Only a few parametric models are closed under PH assumption, the most common of which is the Weibull that accommodates only monotone hazard functions.

The main objective of this thesis is to develop flexible and parsimonious parametric models which are capable of adequately describing different shapes of hazard function. In particular, we propose a generalization of the log-logistic distribution that belongs to the PH family. It has properties similar to those of log-logistic, and approaches the Weibull in the limit. These features enable it to handle both monotone and unimodal (inverse U-shape) hazard functions. Applications to four data sets and a simulation study revealed that the model could potentially be very useful in adequately describing different types of time-to-event data.

The generalized log-logistic PH model naturally accommodates monotone decreasing and unimodal hazard functions, and has the ability to model increasing hazard shapes satisfactorily. However, it is not flexible enough to deal with bathtub-shaped hazard functions. This type of shape is widely used to describe data in medical research and reliability engineering. Motivated by this, we propose a more general parametric proportional hazards model by modifying the Kumaraswamy Weibull (KumW) distribution. The model is parsimonious and flexible in the sense that it accommodates all four standard shapes of the hazard function at the small cost of estimating only three distributional parameters. We also consider two commonly encountered problems in survival analysis which require further extension of the standard PH models. More specifically, we propose methods for recurrent event data analysis and joint modeling as described below.

In biomedical studies and clinical trials, the individuals under study may experience multiple

events over time. Such processes are called recurrent event processes, and the data generated by such processes are called recurrent event data. We propose a parametric recurrent event model, formulated using our MKumW distribution. Specifically, we consider the Poisson process formulation, with the baseline intensity function modeled parametrically.

Another problem considered in this study is joint modeling. In many longitudinal studies, a longitudinal response is observed along with an observation of the time to the occurrence of an event; the event can be timed from the beginning of an observation period, resulting in survival or time-to-event data. A typical goal in such studies is to investigate the effects of the longitudinal response (internal covariate for the event process) on the development of the event. The motivating idea behind the joint modeling techniques is to couple the time-to-event model with the longitudinal model through shared random effects. Although the Cox PH model is appealing to analyze standard survival data mainly due to its robustness property, the use of the Cox PH in joint modeling usually leads to an underestimation of the standard errors of the parameter estimates. Therefore, most methods for joint modeling are based on parametric response distributions. We propose a joint modeling framework based on our MKumW distribution. The novelty lies in formulating a hierarchical model based on the MKumW distribution, proposing a Bayesian approach for statistical inference, and computationally intensive Bayesian implementation of the methodology in the statistical software WinBUGS.

In this thesis, we propose two parametric PH models for time-to-event data, and develop theory for statistical inference. As demonstrated, the proposed models are fairly flexible and parsimonious, and can be valuable in survival analysis theory and applications. Perhaps the most important contribution of this thesis involves further extension of one of the proposed models to recurrent event data analysis and joint modeling of longitudinal and time-to-event data.

We have published one article based on the generalized log-logistic PH model in the *Journal of Statistical Distributions and Applications*. We intend to publish at least two more articles out of this thesis: the focus of one article will be on statistical methodologies for recurrent event and

joint modeling based on the MKumW distribution, and another article could be on the software implementation of the proposed models, possibly in a journal in computational statistics.

Acknowledgements

I am, humbly, grateful to Almighty Allah, the Most Gracious and The Most Merciful, for bequeathing His countless blessings and enabling me for what I never be able. I do acknowledge my earnest deference for the Holy Prophet Hazart Muhammad (P.B.U.H).

I would like to express my deep gratitude and appreciation to my supervisor, Dr. Shahedul A. Khan, who has the merit of teaching me how to do research and who has transmitted to me his enthusiasm for this activity. His willingness to share his ideas in research problems and energy he put in advising my thesis work is highly obliged.

I would like to thank Dr. Cendy Feng, Dr. Juxin Liu and Dr. Nazeem Muhajarine for serving on my advisory committee.

Special Thanks to Bahauddin Zakariya Univsersity, Multan, Pakistan and the Department of Mathematics and Statistics, University of Saskatchewan, Canada for permitting me to indulge with this task.

I offer my regards and blessings to all of those who supported me in any respect during the completion of the thesis.

Most importantly, I am deeply indebted to my family, who missed me back home and counted down every single day throughout the period of study. Finally, a distinctive thank is deserved for the two most precious gifts of my life, my son Muhammad Ibsham Hassan Khan and my daughter Filza Hassan Khan, along with their great father, my husband Jawad Hassan Khan for their unconditional love, sacrifice, encouragement and support- spiritual, emotional, intellectual, financial and otherwise.

This dissertation is dedicated to my mother (Safia Hayat), the memory of my father-in-law (Najeeb-ullah-Khan)

Who filled our days with rainbow lights,
Fairytales and sweet dream nights,
A kiss to wipe away our tears,
Thank you for your tender care,
For deep warm hugs and being there.

CONTENTS

List of Tables

List of Figures

Chapter 1

INTRODUCTION

Analysis of survival data encompasses a variety of statistical techniques involving a positive-valued random variable. Typically, the random variable is the time to the occurrence of an event of interest, commonly known as survival time or lifetime or failure time. The general setup involves a follow-up study, where the observational units are followed over time until the event occurs. (For convenience, we will always use the term "individual" or "subject" to refer to an observational unit in this thesis). The event is commonly known as the survival event, and the outcome or response variable of interest is the survival time (Klein and Moeschberger 2003).

In practice, survival or time-to-event data contain both complete and incomplete observations: the occurrence of the event within the study period leads to a complete observation in the sense that the survival time is observed, whereas nonoccurrence of the event during the observation period leads to only partial information (i.e., the exact survival time is unknown). Incomplete observations arising due to the nonoccurrence of the event within the study period, drop out of a participant or lost to follow-up are called censored observations. Another common feature of survival data is truncation, referring to a late entry of an individual in the study who is then followed until the event occurs (Lee and Wang 2003). Censoring and truncation are two common traits of survival data which make them dissimilar from the standard statistical data sets. These two phenomena also give rise to the need for special statistical techniques to properly handle survival data.

Data sets on failure times typically contain not only the recorded information on the time to an event (T) and censoring status, but also information of explanatory variables (covariates).

As a result, it is of particular interest to develop models to characterize the relationship between the response, T, and one or more covariates which are thought to affect some features of the distribution of T. Therefore, regression models play a very important role in survival analysis, which can be formulated with or without assuming a probability distribution for survival times; distributional assumptions on survival times lead to parametric models, whereas distribution-free assumptions lead to semi-parametric or non-parametric methods (Kalbfleisch and Prentice 2002).

The focus of this thesis is on parametric regression models for survival data in the presence of censored observations, though we will also consider semi-parametric and non-parametric methods for model comparison. In particular, the main objective is to develop flexible models to analyze different types of time-to-event data.

1.1 Background of the Study

There are two popular approaches for regression analysis of survival data: accelerated failure time (AFT) models and proportional hazard (PH) models (Kalbfleisch and Prentice 2002; see also Chapter 2). In order to conceptualize these two approaches, let $S(t) = P(T > t)$ be the probability of an individual surviving beyond time t (commonly known as survivor function), $\mathbf{z}$ be a vector of covariates and $\boldsymbol{\beta}$ the corresponding vector of regression parameters. Also, let $S(t;\mathbf{z}) = P(T > t|\mathbf{z})$ be the survivor function given the covariates $\mathbf{z}$, and $S_0(t)$ the baseline survivor function (i.e., the survival function for an individual with all covariates equal to zero). An AFT model assumes

$$S(t;\mathbf{z}) = S_0(te^{\mathbf{z}'\boldsymbol{\beta}}), \tag{1.1}$$

so the survivor function with covariates $\mathbf{z}$ is the same as survivor function with covariates $\mathbf{z} = 0$ accelerated by the factor $\exp(\mathbf{z}'\boldsymbol{\beta})$ (see Chapter 2 for details), hence the name of the model (Wei 1992). On the other hand, a PH model assumes

$$S(t;\mathbf{z}) = [S_0(t)]^{e^{\mathbf{z}'\boldsymbol{\beta}}}, \tag{1.2}$$

so the survivor function with covariates $\mathbf{z}$ is the baseline survivor raised to a power of $\exp(\mathbf{z}'\boldsymbol{\beta})$ (Lehmann 1953; see Chapter 2 for details). PH models are widely used not only for typical survival data analysis, but also for recurrent event data analysis and joint modeling of longitudinal and time-to-event data. A key reason for the popularity of a PH model is that the regression coefficients have relative risk interpretation. In particular, for a PH model with p covariate $\mathbf{z} = (z_1, z_2, \ldots, z_p)'$, β_j represents the change in the log of the hazard ratio relative to a unit change in z_j, holding all other covariates constant. Thus, $\exp(\beta_j)$ is the hazard ratio (or relative risk) for the effect of z_j, adjusted for the other variables. For example, suppose that $\exp(\beta_j) = 0.3$, where z_j represents treatment status with $z_j = 1$ for a new treatment and $z_j = 0$ for placebo. This suggests that if an individual gets the new treatment, it will reduce the event hazard risk 70% controlling for other factors.

A parametric form for the baseline survivor function in (1.2) leads to a parametric PH model, whereas an arbitrary unspecified baseline survivor function leads to the semi-parametric Cox PH model (Cox 1972). Note that AFT and PH models can also be expressed in terms of hazard functions (see Chapter 2). The Cox model is the most popular in survival analysis mainly because of two reasons: (a) no assumption is required about the probability distribution of survival times, and (b) it usually fits the data well no matter which parametric model is appropriate. In contrast, distributional assumption is required for a fully parametric PH model (Lawless 2003, Klein and Moeschberger 2005). This also leads to the added requirement of checking the appropriateness of the chosen distribution. Nevertheless, as demonstrated by Efron (1977) and Oakes (1977), parametric models lead to more efficient estimates than Cox model under certain conditions. More specifically, if the distributional assumption is valid, a parametric model leads to smaller standard errors of the estimates than would be in the absence of a distributional assumption (Collett 2003). Moreover, the use of Cox PH model in joint modeling of time-to-event and longitudinal data (Wulfsohn and Tsiatis 1997) usually leads to an underestimation of the standard errors of the parameter estimates (Hsieh et al. 2006, Rizopoulos 2012), and therefore most methods for joint modeling are based on parametric response distributions (Hwang and Pennell 2014). Regarding the choice

between a parametric and Cox PH model, Hjort (1992) mentioned that an adequate parametric model could lead to more precise estimates of the survival probabilities and related quantities, and Nardi and Schemper (2003) suggested to use a richer parametric model or simply the Cox model in case of an unsatisfactory result of the chosen probability distribution. In fact, since a fully specified model is often more reliable for analyzing complex data structure and processes, Cox himself expressed his preference towards using parametric models (Reid 1994).

The most commonly used parametric time-to-event models are the Weibull, log- logistic and log-normal distributions. The log-logistic and log-normal distributions belong to the AFT family, and are useful in modeling non-monotone failure/ hazard rates (Lawless 2003). Note that the log-logistic also accommodates decreasing hazard functions. Only a few parametric models are closed under PH assumption, the most common of which is the Weibull that accommodates only monotone hazard functions. In fact, Weibull is the only distribution that is closed under both AFT and PH families (Kalbfleisch and Prentice 2002). Mudholkar et al. (1996) proposed a generalization of the Weibull distribution which permits parametric PH regression modeling. It is a three-parameter distribution and is capable of modeling both monotone and non-monotone hazard functions. One difficulty with this model is that it is nonregular (the support depends on some parameters) for increasing and bathtub-shaped hazard functions, and therefore the standard maximum likelihood asymptotics do not hold.

In summary, the facts which have motivated us to work on parametric PH models are as follows.

- Under certain conditions, a parametric model may lead to more precise estimates compared to non-parametric or semi-parametric methods.
- Only a few parametric models are closed under PH assumption, none of which is flexible enough to describe a wide variety of time-to-event data.
- Parametric PH models are usually preferred in joint modeling of longitudinal and time-to-event data.

Based on these grounds, we now present the objectives of this study.

1.2 Objectives

The main objective of this study is to develop flexible parametric PH models for time-to-event data. We have also tailored our work for the analysis of recurrent event data using the maximum likelihood method and joint modeling of longitudinal and time-to-event using a Bayesian approach. Our main methodological contribution involves extension of the existing models to accommodate different types of hazard function, so that these models can used to adequately fit a wide variety of time-to-event data. The specific objectives are summarized as follows.

1. Formulating flexible parametric distributions which are capable of describing different types of hazard shapes for time-to-event data.
2. Deriving statistical properties of the proposed distributions.
3. Formulating regression models for time-to-event data based on the proposed distributions.
4. Developing asymptotic theory for statistical inference.
5. Evaluating the performance of the proposed models by
   (a) the conventional approach of simulation, and
   (b) comparing the fits of the proposed models with other commonly used survival models in analyzing different types of time-to-event data.
6. Tailoring the proposed method to recurrent event data analysis using the maximum likelihood method, and joint modeling of longitudinal and time-to-event data using a Bayesian approach.
7. Writing computer codes for practical implementation of the proposed techniques.

In the following section 1.3, we briefly elaborate our research contributions.

1.3 Contributions

The log-logistic distribution has wide applications in analyzing survival data. The model is closed under both multiplication of failure time and proportionality of odds. However, it is not a propor-

tional hazard (PH) model. We have developed a simple extension of the log-logistic distribution which is closed under the PH relationship. The proposed generalized log-logistic model (GLL) is a three-parameter distribution, and has characteristics similar to those of the log-logistic model. Moreover, it approaches Weibull in the limit. These features enable the proposed model to satisfactorily handle both monotone (increasing and decreasing) and non-monotone (unimodal or inverse U-shaped) hazard functions. An article based on this work has been published in the *Journal of Statistical Distributions and Applications* (https://doi.org/10.1186/s40488-016-0054-z).

The second distribution we propose is a more flexible model as compared to the generalized log-logistic distribution, formulated on the Kumaraswamy Weibull (KumW) distribution (Cordeiro et al. 2010). It accommodates both monotone (increasing and decreasing) and nonmonotone (unimodal and bathtub shape) hazard functions at the small cost of estimating only one additional parameter compared to Weibull PH model. Comparative studies based on real and simulated data reveal that the model can be valuable in adequately describing different types of time-to-event data. We then develop methods for recurrent event data analysis and joint modeling of time-to-event and longitudinal data. An article about this distribution and its application to recurrent event data analysis and joint modeling is in progress.

In summary, the flexibility provided by the proposed models can be very useful in adequately describing different types of time-to-event data. These models can also be valuable in joint modeling, as semi-parametric model with an unspecified baseline hazard (e.g., Cox model) may lead to inefficient estimates of the joint model parameters.

1.4 Organization of the Thesis

We proceed in Chapter 2 with a review of the mathematical foundation of modeling time-to-event data. There, we also preview the commonly used models in survival analysis, including the log-logistic, Weibull and Cox PH models. We present our work on generalizing the log-logistic dis-

tribution in Chapter 3. Specifically, we describe the proposed distribution, formulation of the PH model based on this distribution, and asymptotic theory for statistical inference. The proposed methodology is then illustrated with application to four data sets and a simulation study. In Chapter 4, we propose another parametric proportional hazards model by modifying the of Kumaraswamy Weibull distribution (Cordeiro et al. 2010), which is parsimonious and flexible in the sense that it accommodates all four standard shapes of the hazard function (increasing, decreasing, unimodal and bathtub shape) at the small cost of estimating only three distributional parameters. A simulation study and real data examples reveal that the proposed model can be valuable in adequately describing different types of time-to-event data. We then extend our modified Kumaraswamy Weibull distribution to model recurrent event data, and to analyze longitudinal and time-to-event data by explicitly taking into account the longitudinal process into the time-to-event model (i.e., joint modeling). These two topics are presented in Chapter 5. In Chapter 6, we summarize our findings with a discussion on the performance of the proposed models. We also discuss some limitations of this study and provide some possible future research directions.

Chapter 2

FOUNDATION OF SURVIVAL DATA ANALYSIS

This chapter reviews the foundation of survival data analysis, including the fundamental concepts and basic methods in modeling survival data in the presence of censored observations. Note that we will not attempt to give an exhaustive description of the techniques, but rather highlight some of the key ideas and elements which form the foundation of this thesis. Since our main focus is on continuous models for survival data with right censoring, methods for discrete analyses will not be presented here. There are several text books emphasizing various aspects of survival data analysis (e.g., Collett 2003, Klein and Moeschberger 2003, Lawless 2003). The interested readers may refer to these books for technical details, including models for discrete analysis, and methods for other types of censoring (e.g., left and interval censoring) and truncated survival data.

2.1 Notation

Let T be a non-negative random variable, representing the lifetimes of individuals in a population. A common feature of time-to-event data is censoring, which occurs when an individual does not experience the event of interest during the targeted period of data collection. Let t be the lifetime or censoring time which we observe at the end of the study. The indicator variable $\delta = I(T = t)$ denotes the status of a particular observation, where $\delta = I(T = t)$ equals 1 if $T = t$ and 0 if $T > t$. This variable is known as censoring indicator, and provides information about whether t is observed ($\delta = 1$) or censored ($\delta = 0$). The vector of p covariates will be denoted by $\mathbf{z} = (z_1, z_2, \ldots, z_p)'$, and the corresponding vector of regression parameters by $\boldsymbol{\beta} = (\beta_1, \beta_2, \ldots, \beta_p)'$.

2.2 Basic Elements of Survival Data Analysis

There are four basic functions which play the pivotal role in modeling survival data: probability density function, survivor function, hazard function and cumulative hazard function. Below in this section, we define these functions for a continuous random variable T.

Survivor Function

The survivor function is defined as the probability that an individual survives at least t units, and is denoted by $S(t)$. Mathematically,

$$S(t) = P(T \geq t). \tag{2.1}$$

$S(t)$ is a monotone, non-increasing and left continuous function, with $S(0) = 1$ and $\lim_{t\to\infty} S(t) = 0$. The graph of $S(t)$ as a function of t is called the survival curve. Note that $S(t) = 1 - F(t)$, where $F(t)$ is the cumulative density function (cdf) of T.

Probability Density Function

The probability density function (pdf) is defined by

$$f(t) = -\frac{d}{dt}S(t) = \lim_{\Delta t\to 0} \frac{Pr(t \leq T < t + \Delta t)}{\Delta t}, \tag{2.2}$$

which is the rate of increase of $1 - S(t)$, so that

$$S(t) = \int_t^\infty f(s)ds. \tag{2.3}$$

Hazard Function

The hazard function, denoted by $h(t)$, is defined as

$$h(t) = \lim_{\Delta t\to 0} \frac{Pr(t \leq T < t + \Delta t | T \geq t)}{\Delta t} = \frac{f(t)}{S(t)}, \tag{2.4}$$

which gives the instantaneous rate per unit time for the event to occur, given that the individual survives up to time t. Note that $h(t)\Delta t$ is the approximate probability of the event to occur in $[t, t+\Delta t)$, given survival up to time t. The hazard function is also commonly known as the hazard rate or failure rate. It is easy to verify that (Lawless 2003)

$$h(t) = -\frac{S'(t)}{S(t)} = -\frac{d}{dt}\log S(t), \tag{2.5}$$

so that

$$\log S(t) = -\int_0^t h(s)ds, \tag{2.6}$$

and hence

$$S(t) = \exp\Big\{-\int_0^t h(s)ds\Big\}. \tag{2.7}$$

It follows that

$$f(t) = h(t)S(t) = h(t)\exp\Big\{-\int_0^t h(s)ds\Big\}. \tag{2.8}$$

Cumulative Hazard Function

The cumulative hazard function, denoted by $H(t)$, is defined as

$$H(t) = \int_0^t h(s)ds, \tag{2.9}$$

which may be interpreted as the expected number of events that occur up to a given time (Collett 2003). The relationship $H(t) = -\log S(t)$ plays an important role to check adequacy of a parametric time-to-event model, and to formulate likelihood functions for censored survival data.

2.3 Non-parametric Methods

An exploratory analysis of time-to-event data typically involves nonparametric methods of estimation of survivor functions and /or cumulative hazard functions. Nonparametric methods are useful

to get insights about the main characteristics of the distribution of T, and are often used to check the appropriateness of a parametric model. In survival analysis, the most widely used nonparametric techniques are the Kaplan-Meier (KM) estimator of survivor functions (Kaplan and Meier 1958) and the Nelson-Aalen estimator of the cumulative hazard functions (Nelson 1972).

2.3.1 Kaplan-Meier Estimator of the Survivor Function

Let $t_1 < t_2 < \ldots < t_k$ be the ordered observed lifetimes from a sample of size n. Also, let r_j be the number of individuals at risk of failure at time $t = t_j$ (commonly known as the risk set), d_j the number of individuals who experience the event at time $t = t_j$, and c_j the number of individuals with censoring times in $[t_j, t_{j+1})$, where $j = 0, 1, \ldots, k$, $t_0 = 0$ and $t_{k+1} = \infty$. Under this setting, it is easy to see that $r_j = d_j + c_j + d_{j+1} + c_{j+1} + \ldots + d_k + c_k$. The KM estimator of the survivor function is given by (see Lawless (2003) for theoretical details)

$$\hat{S}(t) = \prod_{j|t_j<t} \left(1 - \frac{d_j}{r_j}\right), \tag{2.10}$$

which implies that the conditional probability of the occurrence of an event at each observed time t_j is the observed conditional relative frequency of the event at t_j (i.e., d_j/r_j). Note that if a censoring time and a lifetime are recorded as equal, the general convention is to regard the censoring time as being infinitesimally larger in the definition of $\hat{S}(t)$. The derivation of the pointwise confidence intervals for $S(t)$ based on the KM method is discussed in Lawless (2003).

2.3.2 Nelson-Aalen Estimator of the Cumulative Hazard Function

The Nelson-Aalen estimator of the cumulative hazard function is given by

$$\hat{H}(t) = \sum_{j|t_j<t} \frac{d_j}{r_j}. \tag{2.11}$$

A plot of $\hat{H}(t)$ versus t is called a cumulative hazard plot, which is frequently used as a diagnostic tool to check the assumption of a parametric model. For further details regarding the estimation of $H(t)$ and cumulative hazard plots, see Nelson (1972) and Aalen (1978).

2.4 Parametric Failure Time Models

Certain probability distributions are extensively used to model time-to-event data (e.g., exponential, Weibull, log-logistic and log-normal). The popularity of these in survival analysis is largely due to (a) model parsimony, (b) straightforwardness of the approach, (c) ability to satisfactorily model data which are commonly encountered in survival analysis, and (d) readily available statistical software packages. This section reviews some continuous distributions used for the analysis of time-to-event data.

2.4.1 Exponential Distribution

The exponential distribution is the simplest model for lifetime data. It has only one parameter, and therefore not flexible enough to describe commonly encountered hazard shapes for time-to-event data. The probability density function, survivor function and hazard function of the exponential distribution are, respectively,

$$f(t) = \rho \exp(-\rho t),$$

$$S(t) = \exp(-\rho t),$$

$$h(t) = \rho,$$

for $t \geq 0$, where $\rho > 0$ is the rate parameter (the scale parameter $\lambda = \rho^{-1}$ is also used in formulating the model); a large value of ρ indicates high risk and short survival, whereas a small value indicates low risk and long survival. The distribution with $\rho = 1$ is called the standard exponential distribution. Since the hazard rate is constant, the exponential distribution often found to be inadequate to describe time-to-event data. This makes the applicability of this distribution fairly limited.

2.4.2 Weibull Distribution

The Weibull distribution (Weibull 1951) is a generalization of the exponential distribution. It has an additional parameter which describes the shape of the hazard function. The probability density function, survivor function and hazard function of the Weibull distribution are, respectively,

$$f(t) = \kappa\rho(\rho t)^{\kappa-1} \exp\{-(\rho t)^{\kappa}\},$$
$$S(t) = \exp\{-(\rho t)^{\kappa}\},$$
$$h(t) = \kappa\rho(\rho t)^{\kappa-1},$$

where $t > 0$ is the support of the distribution. The parameters of the distribution are $\rho > 0$ and $\kappa > 0$, where ρ is the rate parameter (the scale parameter $\lambda = \rho^{-1}$ is often used in place of ρ) and κ is the shape parameter. The Weibull hazard function is monotone increasing when $\kappa > 1$, decreasing for $\kappa < 1$, and constant for $\kappa = 1$ (the Weibull distribution reduces to exponential for $\kappa = 1$). Note that the Weibull distribution does not accommodate non-monotone hazard functions.

If T has a Weibull distribution, then $Y = \log T$ has an extreme value distribution. The extreme value distribution plays an important role in regression modeling (see Section 2.6.1). The probability density function and survivor function of the extreme value distribution are, respectively,

$$f(y) = \tau^{-1} \exp\left\{\frac{y-\alpha}{\tau} - \exp\left(\frac{y-\alpha}{\tau}\right)\right\},$$
$$S(y) = \exp\left\{-\exp\left(\frac{y-\alpha}{\tau}\right)\right\},$$

where $\alpha = -\log\rho$ and $\tau = \kappa^{-1}$. Here, α and τ are the location and scale parameters, respectively. The distribution of $(Y-\alpha)/\tau$ is called the standard extreme value distribution for which the location and scale parameters are 0 and 1, respectively.

The Weibull distribution is perhaps the most widely used distribution in survival analysis. The model is fairly flexible (accommodates monotone hazard shapes), and has simple expressions for $f(t)$, $S(t)$ and $h(t)$. It is particularly popular in engineering applications, as it has the ability to

model different types of reliability data exhibiting either skewed or symmetric distributional shape. Examples of applications include modeling lifetimes of electrical and mechanical components, automobile components and ceramic capacitors.

2.4.3 Log-Logistic Distribution

The log-logistic distribution is particularly useful to model unimodal (i.e., non-monotone) hazard functions. The probability density function, survivor function and hazard function of the log-logistic distribution are, respectively,

$$f(t) = \frac{\kappa\rho(\rho t)^{\kappa-1}}{\left[1 + (\rho t)^{\kappa}\right]^2},$$

$$S(t) = \frac{1}{1 + (\rho t)^{\kappa}},$$

$$h(t) = \frac{\kappa\rho(\rho t)^{\kappa-1}}{1 + (\rho t)^{\kappa}},$$

where $t > 0$ is the support of the distribution, and $\rho > 0$ and $\kappa > 0$ are the parameters, where ρ is the rate parameter and κ is the shape parameter. It is easy to verify that the hazard function of the log-logistic distribution is monotone decreasing for $\kappa \leq 1$, and unimodal for $\kappa > 1$.

If T has a log-logistic distribution, then $Y = \log T$ has a logistic distribution. The logistic distribution is used to formulate regression models for time-to-event data (Section 2.6.1). The probability density function and survivor function of the logistic distribution are, respectively,

$$f(y) = \frac{\exp\left(\frac{y-\alpha}{\tau}\right)}{[1 + \exp\left(\frac{y-\alpha}{\tau}\right)]^2},$$

$$S(y) = 1 - \frac{\exp\left(\frac{y-\alpha}{\tau}\right)}{[1 + \exp\left(\frac{y-\alpha}{\tau}\right)]},$$

where $\alpha = -\log\rho$ is the location parameter and $\tau = \kappa^{-1}$ is the scale parameter. The distribution of $(Y - \alpha)/\tau$ is called the standard logistic distribution.

The log-logistic distribution is widely used to describe the course of a disease where mortality reaches a peak after some finite period, and then slowly declines (Bennett 1983). For example, the log-logistic model can be used to describe the lifetimes of breast cancer patients (peak mortality of breast cancer patients occurs after about three years (Langlands et al. 1979)).

2.4.4 Log-normal Distribution

The log-normal distribution is another popular model to describe non-monotone hazard functions. The probability density function, survivor function and hazard function of the log-normal distribution are, respectively,

$$f(t) = \frac{1}{t\tau\sqrt{2\pi}} \exp\Big\{ -\frac{1}{2}\Big(\frac{\log \rho t}{\tau}\Big)^2\Big\},$$

$$S(t) = 1 - \Phi\Big(\frac{\log \rho t}{\tau}\Big),$$

$$h(t) = \frac{f(t)}{S(t)},$$

where $t > 0$ is the support of the distribution, $\Phi(\cdot)$ is the standard normal cdf, and $\tau > 0$ and $\rho > 0$ are the parameters. Note that if T is log-normal with parameters ρ and τ, then $Y = \log T$ is normal with mean $\alpha = -\log \rho$ and variance τ^2 (Lawless 2003).

The shape of the hazard functions for log-logistic and log-normal distributions are very similar when the log-logistic shape parameter $\kappa > 1$: $h(t) = 0$ at $t = 0$, increases to a maximum as t increases, and then decreases after reaching a threshold time, approaching 0 as $t \to \infty$. As described in Section 2.4.3, this type of hazard shape arises in many applications, including survival of cancer patients after treatment and product failures caused by chemical reactions or corrosions.

2.4.5 Gamma Distribution

The random variable T has a gamma distribution if

$$f(t) = \frac{\rho^{\kappa} t^{\kappa-1} \exp\{-\rho t\}}{\Gamma(\kappa)} \quad t > 0,$$

where $\rho^{-1} > 0$ is a scale parameter and $\kappa > 0$ is a shape parameter. The gamma distribution includes the exponential distribution as a special case ($\kappa = 1$). The survivor function of the gamma distribution is given by

$$S(t) = 1 - I_{\kappa}(\rho t),$$

where $I_{\kappa}(t) = \int_0^t \frac{u^{\kappa-1} \exp\{-u\}}{\Gamma(\kappa)} du$ is the incomplete gamma function. The hazard function is given by

$$h(t) = \frac{\rho^{\kappa} t^{\kappa-1} \exp\{-\rho t\}}{\Gamma(\kappa)\{1 - I_{\kappa}(\rho t)\}},$$

which is monotone increasing for $\kappa > 1$ ($h(t) = 0$ at $t = 0$, and $h(t) \to \rho$ as $t \to \infty$), and monotone decreasing for $0 < \kappa < 1$ ($h(t) \to \infty$ as $t \to 0$, and $h(t) \to \rho$ as $t \to \infty$).

The gamma distribution is not used as much as the Weibull, log-logistic and log-normal distributions in survival analysis, partly because the survivor and the hazard functions of the gamma distribution are intractable from a computational point of view. This may also lead to computational difficulties in the maximum likelihood estimation of the parameters. Nevertheless, the gamma distribution has been used to model lifetimes of technical systems with repeated repairing after failure, rainfall data in meteorology, and insurance claims and loan data in business (Thom 1958).

2.4.6 Generalized Gamma Distribution

The probability density function of the generalized gamma distribution is

$$f(t) = \frac{|\lambda|(\lambda^{-2})^{\lambda^{-2}}}{\sigma t \Gamma(\lambda^{-2})} \exp\left\{\frac{\lambda(\frac{\log t - \alpha}{\sigma}) - e^{\lambda(\frac{\log t - \alpha}{\sigma})}}{\lambda^2}\right\},$$

where $\infty < \alpha < \infty$, $\sigma > 0$ and $\lambda > 0$ are parameters. This parameterization is preferred to the original parameterization of the generalized gamma by Stacy (1962), since it is more numerically stable (Jackson 2016). The generalized gamma distribution includes all four basic shapes of the hazard function: increasing for $0 < \sigma < 1$ and $\lambda \leq 1/\sigma$, decreasing for $\sigma > 1$ and $\lambda \geq 1/\sigma$, unimodal for $\lambda < \min(\sigma, 1/\sigma)$, and bathtub shape for $\lambda > \max(\sigma, 1/\sigma)$. The generalized gamma distribution also includes the following special cases: Weibull when $\lambda = 1$, $\sigma = 1/\kappa$ and $\alpha = -\log\rho$, log-normal when $\lambda = 0$ and $\alpha = -\log\rho$, and gamma when $\lambda = \sigma$. The generalized gamma distribution has applications in many fields, including income data analysis in economics (Kleiber and Kotz 2003) and flood frequency data in civil engineering (Pham and Almhana 1995).

2.4.7 Exponentiated Weibull Distribution

The exponentiated Weibull distribution (Mudholkar and Srivastava 1993) is a generalization of the two parameter Weibull distribution described in Section 2.4.2. The probability density function, hazard function and survivor function of the exponentiated Weibull distribution are, respectively,

$$f(t) = \kappa\rho\gamma(\rho t)^{\kappa-1}\Big(1 - \exp\{-(\rho t)^{\kappa}\}\Big)^{\gamma-1} \exp\{-(\rho t)^{\kappa}\},$$

$$h(t) = \frac{\kappa\rho\gamma(\rho t)^{\kappa-1}(1 - \exp\{-(\rho t)^{\kappa}\})^{\gamma-1} \exp\{-(\rho t)^{\kappa}\}}{1 - (1 - \exp\{-(\rho t)^{\kappa}\})^{\gamma}},$$

$$S(t) = 1 - (1 - \exp\{-(\rho t)^{\kappa}\})^{\gamma},$$

where $t > 0$ is the support of the distribution, $\rho > 0$ is a rate parameter, and $\kappa > 0$ and $\gamma > 0$ are shape parameters. Note that $\gamma = 1$ reduces the exponentiated Weibull to the two-parameter Weibull distribution. Mudholkar and Srivastava (1993) showed that the hazard function is (a) monotone increasing for $\kappa \geq 1$ and $\kappa\gamma \geq 1$, (b) monotone decreasing for $\kappa \leq 1$, and $\kappa\gamma \leq 1$, (c) unimodal for $\kappa < 1$ and $\kappa\gamma > 1$, and (d) bathtub-shaped for $\kappa > 1$ and $\kappa\gamma < 1$.

The exponentiated weibull distribution has demonstrated considerable potential in describing

different types of time-to-event data. The model is flexible and parsimonious, as it can accommodate both monotone and non-monotone hazard functions at the cost of estimating only three parameters. It has been successfully applied to model bathtub failure rates of electrical devices (Mudholkar and Srivastava 1993), bus-motor failure data (Mudholkar et al. 1996), and cancer survival data (Khan 2017). A simple two parameter distribution with bathtub shape or increasing hazard rate has been proposed by Chen (2000). An extension of Chen's family of distribution and exponentiated Chen distribution provides an alternative to generalized Weibull and exponentiated Weibul families (Chaubey and Zhang 2015).

2.5 Maximum Likelihood Estimation

Let there be n individuals with lifetimes denoted by $T_1, T_2, \ldots, T_n$. Assuming that the data are subject to right censoring, we observe $t_i = \min(T_i, C_i)$, where $C_i > 0$ corresponding to a potential censoring time for individual i. Letting $\delta_i = I(T_i \leq C_i)$ that equals 1 if $T_i \leq C_i$ and 0 otherwise , the observed data for individual i consist of $\{t_i, \delta_i\}, i = 1, 2, \ldots, n$, where t_i is a lifetime or censoring time according to whether $\delta_i = 1$ or 0 respectively. We assume non-informative censoring, where the distribution of survival times provides no information about the distribution of censoring times, and vice versa. Note that the assumption of non-informative censoring is justifiable when censoring is random (i.e failure rates for censored and uncensored observations who remain in the risk set are assumed equal) and / or independent (i.e., censoring is assumed random within any subgroup of interest); see Kleinbaum and Klein (2012) for technical details.

Under non-informative censoring, t_i and δ_i are random variables with $P(t_i = c_i, \delta_i = 0) = P(T_i > C_i) = S(t_i)$ and $P(t_i, \delta_i = 1) = f(t_i)$. Then, the joint p.d.f of t_i and δ_i is $[f(t_i)]^{\delta_i}[S(t_i)]^{(1-\delta_i)}$, which is the contribution of the i^{th} individual to the likelihood function. Thus, individual i contributes $f(t_i)$ to the likelihood function if an event occurs at time t_i, and contributes $S(t_i)$ if the individual is censored at t_i. Combining the information from the censored observations, we obtain

the likelihood fucntion

$$L(\boldsymbol{\theta}) = \prod_{i=1}^{n} [f(t)]^{\delta_i} [S(t)]^{1-\delta_i}, \tag{2.12}$$

where $\boldsymbol{\theta}$ is a vector of parameters characterising the distribution of T_i. Using (2.8), the likelihood function can also be written as

$$L(\boldsymbol{\theta}) = \prod_{i=1}^{n} \Big[h(t)\Big]^{\delta_i} \exp\Big\{ - \int_0^t h(s)ds \Big\}. \tag{2.13}$$

An iteravtive optimization procedure (e.g., Newton-Raphson algorithm) can be used to obtain the maximum likelihood estimate $\hat{\boldsymbol{\theta}}$ of $\boldsymbol{\theta}$. Hypothesis testing and interval estimates of the model parameters can proceed under the approximate normality of the maximum likelihood estimators (Lawless 2003).

2.6 Regression Models for Time-to-Event Data

Statistical analysis is often needed to prepare summary of data for prediction. One way to achieve this is to search for a theoretical model which adequately fits the observed data and identify the covariates which are significantly associated with the response. There are two popular approaches for the regression analysis of time-to-event data: accelerated failure time (AFT) model and proportional hazards (PH) model. The formulation of these models is based on a function $\psi(\mathbf{z}'\boldsymbol{\beta})$ such that (a) $\psi(\mathbf{z}'\boldsymbol{\beta}) > 0$, (b) $\psi(\mathbf{z}'\boldsymbol{\beta})$ is one-to-one monotone function, and (c) $\psi(0) = 1$. The most natural choice for $\psi(\mathbf{z}'\boldsymbol{\beta})$ is the exponential function $\exp(\mathbf{z}'\boldsymbol{\beta})$. Below in this section, we describe the AFT and PH models assuming $\psi(\mathbf{z}'\boldsymbol{\beta}) = \exp(\mathbf{z}'\boldsymbol{\beta})$.

2.6.1 The Accelerated Failure Time Model

According to the AFT model, the covariates act multiplicative on survival time. Taking $\psi(\mathbf{z}'\boldsymbol{\beta}) = \exp(\mathbf{z}'\boldsymbol{\beta})$, the hazard function under the AFT assumption is (Kalbfleisch and Prentice 2002)

$$h(t;\mathbf{z}) = h_0(te^{-\mathbf{z}'\boldsymbol{\beta}})e^{-\mathbf{z}'\boldsymbol{\beta}}, \tag{2.14}$$

where $h_0(\cdot)$ is the baseline hazard function (i.e., hazard for an individual with $\mathbf{z} = \mathbf{0}$). The survivor function can be expressed as

$$S(t;\mathbf{z}) = \exp\left\{-\int_0^t h_0(ue^{-\mathbf{z}'\boldsymbol{\beta}})e^{-\mathbf{z}'\boldsymbol{\beta}}du\right\} = \exp\{-H_0(te^{-\mathbf{z}'\boldsymbol{\beta}})\} = S_0(te^{-\mathbf{z}'\boldsymbol{\beta}}), \tag{2.15}$$

where $H_0(\cdot)$ and $S_0(\cdot)$ are the baseline cumulative hazard function and survivor functions, respectively. Using (2.14) and (2.15), the probability density function is

$$f(t;\mathbf{z}) = f_0(te^{-\mathbf{z}'\boldsymbol{\beta}})e^{-\mathbf{z}'\boldsymbol{\beta}}. \tag{2.16}$$

The survivor function (2.15) of the AFT model can be interpreted as follows: the probability of an individual (with covariates $\mathbf{z}$) surviving to time t is the same as the probability of a reference individual surviving to time $te^{-\mathbf{z}'\boldsymbol{\beta}}$. We also see from (2.14) - (2.16) that the covariates act multiplicatively on time so that their effect is to accelerate or decelerate (depending on the value of $\boldsymbol{\beta}$) the time to failure, hence the name of the model.

Now, let T_0 be a random variable corresponding to the lifetime when $\mathbf{z} = 0$, so that the survivor function of T_0 is of the form $S_0(\cdot)$. Based on this definition, we have $T_0 = Te^{-\mathbf{z}'\boldsymbol{\beta}}$ from (2.15). The AFT model can then be expressed as

$$\begin{aligned}
&\log T - \mathbf{z}'\boldsymbol{\beta} = \log T_0 \\
&\Rightarrow \frac{\log T - (\beta_0 + \mathbf{z}'\boldsymbol{\beta})}{\tau} = \frac{\log T_0 - \beta_0}{\tau} \\
&\Rightarrow \frac{\log T - (\beta_0 + \mathbf{z}'\boldsymbol{\beta})}{\tau} = W \\
&\Rightarrow Y = \beta_0 + \mathbf{z}'\boldsymbol{\beta} + \tau W,
\end{aligned} \tag{2.17}$$

where $Y = \log T$, $\tau > 0$ is a scale parameter, and $W = \frac{\log T_0 - \beta_0}{\tau}$ is the error component.

The error term W in (2.17) is assumed to follow a standard distribution such as the extreme value, normal or logistic. These lead to Weibull, log-normal and log-logistic models for T, respectively. As an example, suppose that T has a Weibull distribution with parameters ρ and κ. Using (2.14), the hazard function with covariate vector $\mathbf{z}$ is

$$h(t;\mathbf{z}) = h_0(te^{-\mathbf{z}'\boldsymbol{\beta}})e^{-\mathbf{z}'\boldsymbol{\beta}} = \kappa\rho^{\kappa}(te^{-\mathbf{z}'\boldsymbol{\beta}})^{\kappa-1}e^{\mathbf{z}'\boldsymbol{\beta}} = \kappa\rho^{\kappa}t^{\kappa-1}e^{-\kappa\mathbf{z}'\boldsymbol{\beta}} = \kappa(\rho e^{-\mathbf{z}'\boldsymbol{\beta}})^{\kappa}t^{\kappa-1}\,, \tag{2.18}$$

which is again the Weibull hazard with $\rho^* = \rho e^{-\mathbf{z}'\boldsymbol{\beta}}$. Using (2.18), the survivor function is

$$S(t;\mathbf{z}) = \exp\Big\{-\left(\rho e^{-\mathbf{z}'\boldsymbol{\beta}}t\right)^{\kappa}\Big\} = \exp\Big\{-[\rho(te^{-\mathbf{z}'\boldsymbol{\beta}})]^{\kappa}\Big\}, \tag{2.19}$$

which is the form $S_0(te^{-\mathbf{z}'\boldsymbol{\beta}})$. It follows that the Weibull family is closed under the AFT relationship, and therefore can be expressed in the form $Y = \beta_0 + \mathbf{z}'\boldsymbol{\beta} + \tau W$. Since T_0 is assumed to follow a Weibull distribution with parameters ρ and κ, $W = \frac{\log T_0 - \beta_0}{\tau}$ has the standard extreme value distribution with $\beta_0 = -\log\rho$ and $\tau = \kappa^{-1}$ (see Section 2.4.2).

Using similar arguments, we can show that the log-normal and log-logistic also belong to the AFT family with the distribution of W being the standard normal and standard logistic, respectively.

Suppose that a censored random sample consisting of data $\{t_i, \delta_i, \mathbf{z}_i\}$, $i = 1, 2, \ldots, n$, is available. Using (2.12), the log-likelihood function can be written as

$$\ell(\boldsymbol{\theta}) = \sum_{i=1}^{n}\delta_i \log[f(t_i;\mathbf{z}_i)] + \sum_{i=1}^{n}(1-\delta_i)\log[S(t_i;\mathbf{z}_i)], \tag{2.20}$$

where $\boldsymbol{\theta}$ is a column vector of parameters. For an AFT model, the pdf of $Y = \log T$ can be expressed as $\frac{1}{\tau}f_0(w)$, where $w = \frac{y-\beta_0-\mathbf{z}'\boldsymbol{\beta}}{\tau}$ (Lawless 2003). Thus, the log-likelihood function can be equivalently written as

$$\ell(\boldsymbol{\theta}) = -\sum_{i=1}^{n}\delta_i\log\tau + \sum_{i=1}^{n}\delta_i\log f_0(w_i) + \sum_{i=1}^{n}(1-\delta_i)\log S_0(w_i), \tag{2.21}$$

which is numerically more convenient to use for the maximum likelihood method. As mentioned in Section 2.5, inference then proceeds under the classical asymptotic maximum likelihood theory.

2.6.2 The Proportional Hazards Model

According to the proportional hazards assumption, the effect of a covariate is to increase or decrease the hazard by a proportionate amount which does not depend on t. Under this assumption, the hazard function with covariate vector $\mathbf{z}$ (fixed/time dependent) is

$$h(t;\mathbf{z}) = h_0(t)e^{\mathbf{z}'\boldsymbol{\beta}}. \tag{2.22}$$

It is clear from (2.22) that the hazard ratio comparing any two specifications of the covariates, say $\mathbf{z}$ and $\mathbf{z}^*$, is

$$\frac{h(t;\mathbf{z})}{h(t;\mathbf{z}^*)} = \exp\left[(\mathbf{z}' - \mathbf{z}^{*\prime})\boldsymbol{\beta}\right], \tag{2.23}$$

which is constant over time. This means that the hazard for one individual is proportional to the hazard for any other individual, where the proportionality constant is independent of time. The survivor function and probability density function for a PH model are, respectively,

$$S(t;\mathbf{z}) = \left[S_0(t)\right]^{\exp(\mathbf{z}'\boldsymbol{\beta})}, \tag{2.24}$$

$$f(t;\mathbf{z}) = f_0(t)e^{\mathbf{z}'\boldsymbol{\beta}}\left[S_0(t)\right]^{\exp(\mathbf{z}'\boldsymbol{\beta})-1}. \tag{2.25}$$

A PH model can be formulated by assuming an arbitrary and unspecified baseline hazard function $h_0(t)$ in (2.22). This leads to the well-known Cox PH model (Cox 1972), which does not rely on any distributional assumption for the outcome variable. A fully parametric PH model can also be formulated by assuming a parametric form for $h_0(t)$. The exponential and the Weibull distribution can be used for this purpose (the log-logistic and log-normal are not closed under the PH relationship). For example, suppose that T has a Weibull distribution with parameters ρ and κ.

Using (2.22), we have

$$h(t;\mathbf{z}) = h_0(t)e^{\mathbf{z}'\boldsymbol{\beta}} = \kappa\rho^{\kappa}t^{\kappa-1}e^{\mathbf{z}'\boldsymbol{\beta}} = \kappa(\rho e^{\mathbf{z}'\boldsymbol{\beta}/\kappa})^{\kappa}t^{\kappa-1}, \tag{2.26}$$

which is again the Weibull hazard with $\rho^* = \rho e^{\mathbf{z}'\boldsymbol{\beta}/\kappa}$. It follows that the Weibull is closed under the PH relationship. In fact, the Weibull model is the only family which is closed under both multiplication of failure time (AFT family) and multiplication of the hazard function (PH family) by an arbitrary nonzero constant (Kalbfleisch and Prentice 2002). Recall that $\rho^* = \rho e^{-\mathbf{z}'\boldsymbol{\beta}}$ for the Weibull AFT model, and therefore the regression coefficients of the Weibull AFT and PH models are related as follows: β_j for the Weibull PH model $= -\kappa \times \beta_j$ for the Weibull AFT model for $j = 1, 2, \ldots, p$ (for a rigorous proof, see Kalbfleisch and Prentice (2002)).

If a parametric PH model is considered, the log-likelihood function can be written using (2.13) as follows:

$$\ell(\boldsymbol{\theta}) = \sum_{i=1}^{n} \delta_i[\log h_0(t_i;\boldsymbol{\zeta}) + \mathbf{z}_i'\boldsymbol{\beta}] - \sum_{i=1}^{n} H_0(t_i;\boldsymbol{\zeta}) \exp(\mathbf{z}_i'\boldsymbol{\beta}), \tag{2.27}$$

where $\boldsymbol{\theta} = (\boldsymbol{\zeta}', \boldsymbol{\beta}')'$, $\boldsymbol{\zeta}$ is a vector of the distributional parameters, and $\boldsymbol{\beta}$ is a vector of the regression coefficients. This can be maximized directly using the Newton-Raphson optimization algorithm.

For the Cox PH model, Cox (1972) proposed a partial likelihood method to estimate $\boldsymbol{\beta}$ without having to consider a parametric form for $h_0(t)$ (i.e., $h_0(t)$ is assumed arbitrary and unspecified). A detailed description of the partial likelihood function is given in many text books (e.g., Collett 2003, Kalbfleisch and Prentice 2002). Here we only provide a comprehensive overview of the partial likelihood method to estimate $\boldsymbol{\beta}$.

Suppose we have n individual under study. Let $t_1, t_2, \ldots, t_n$ denote their failure/censoring times, and $\delta_1, \delta_2, \ldots, \delta_n$ denote the corresponding censoring indicators. The observed data for individual i consist of $\{t_i, \delta_i, \mathbf{z}_i\}, i = 1, 2, \ldots, n$.

Let $t_{(1)} < t_{(2)}, \ldots, t_{(K)}$ denote the unique ordered observed failure times, $i_{(k)}$ denotes the individual with the failure time $t_{(k)}$, $k = 1, 2, \ldots, K$, and $R(t) = \{i : t_i \geq t\}$ denotes the set of all individual at

risk for failure at time t, commonly known as the risk set. Using (2.22), the conditional probability that the individual $i_{(k)}$ failed at time $t_{(k)}$ given one subject at risk failed at that time is

$$\frac{h_0(t_{(k)})\exp(\mathbf{z}'_{i_{(k)}}\boldsymbol{\beta})}{\sum_{l\in R(t_{(k)})} h_0(t_{(k)})\exp(\mathbf{z}'_l\boldsymbol{\beta})} = \frac{\exp(\mathbf{z}'_{i_{(k)}}\boldsymbol{\beta})}{\sum_{l\in R(t_{(k)})}\exp(\mathbf{z}'_l\boldsymbol{\beta})}. \tag{2.28}$$

Taking the product of such terms over all $k = 1, 2, \ldots, K$, we obtain

$$L(\boldsymbol{\beta}) = \prod_{k=1}^{K} \frac{\exp(\mathbf{z}'_{i_{(k)}}\boldsymbol{\beta})}{\sum_{l\in R(t_{(k)})}\exp(\mathbf{z}'_l\boldsymbol{\beta})}. \tag{2.29}$$

This is not a likelihood function in the usual sense, because it is the product of the conditional probabilities, where the conditioning event is changing over time. Cox (1972) argued that this should behave roughly like a likelihood function, and could be used as a basis for inference for $\boldsymbol{\beta}$. It is called the partial likelihood function for the Cox PH model. Note that (2.29) can also be written as

$$L(\boldsymbol{\beta}) = \prod_{i=1}^{n}\left[\frac{\exp(\mathbf{z}'_i\boldsymbol{\beta})}{\sum_{l\in R(t_{(i)})}\exp(\mathbf{z}'_l\boldsymbol{\beta})}\right]^{\delta_i}, \tag{2.30}$$

and the partial log-likelihood as

$$\ell(\boldsymbol{\beta}) = \sum_{i=1}^{n}\delta_i\left[(\mathbf{z}'_i\boldsymbol{\beta}) - \log\left(\sum_{l\in R(t_{(i)})}\exp(\mathbf{z}'_l\boldsymbol{\beta})\right)\right]. \tag{2.31}$$

The maximum likelihood estimate of $\boldsymbol{\beta}$ can have obtained by maximizing (2.31) using Newton-Raphson iteration or other optimization methods.

2.7 Model Diagnostics

Checking the adequacy of a model in describing a dataset is an essential part in any statistical analysis. It is generally recommended to assess the adequacy before using a model for decision-

making purposes. Ideally, we would like our model to be flexible and parsimonious, with the ability to fit a wide range of data satisfactorily. Thus, an assessment of the quality of a fit and adherence to model assumptions are as important as model development in any statistical analysis. Many of the model diagnostic procedures are based on graphical assessment. For example, in a univariate analysis without covariates, we may examine the plot of parametric and nonparametric estimates of the survival function, superimposed on the same graph. If $S(t;\hat{\boldsymbol{\theta}})$ and $\hat{S}(t)$ are the estimates of the survivor functions based on the parametric model of interest and Kaplan-Meier method, respectively, then $S(t;\hat{\boldsymbol{\theta}})$ as a function of t should be close to $\hat{S}(t)$ if the parametric model is adequate. For models with covariates, commonly used diagnostic tools include analyses of hazard based residuals, martingale residuals, score residuals and Schoenfeld residuals to check the quality of a fit and the validity of the underlying model assumptions. In addition, statistical tests and goodness of fit criteria can be used to check the proportionality assumption of a PH model, and to compare the fits of competing models with a view towards identifying a model with the fewest parameters (i.e., parsimonious) that provides an adequate fit to the data. Below in this section, we present some of these techniques that are considered in this thesis (see Lawless (2003) for a detailed description of these methods).

2.7.1 Hazard Based Residuals

Hazard based residuals are defined based on the cumulative hazard function $H(t;\boldsymbol{\theta})$, where $\boldsymbol{\theta}$ is a vector of parameters associated with the model. Since the cumulative hazard function is a monotonic function, $H(T;\boldsymbol{\theta})$ is simply a monotonic transforamtion of T. Thus, if $\boldsymbol{\theta}$ is known, $H(T;\boldsymbol{\theta})$ can be considered as a random variable. Using (2.1), (2.7) and (2.9), we have

$$P(T \geq t;\boldsymbol{\theta}) = P[H(T;\boldsymbol{\theta}) \geq H(t;\boldsymbol{\theta})] = S[H(t;\boldsymbol{\theta})] = \exp\{-H(t;\boldsymbol{\theta})\}, \tag{2.32}$$

which leads to the conclusion that $H(T;\boldsymbol{\theta})$ has an exponential distribution with rate one. With $\boldsymbol{\theta}$

known and in the absence of censoring, we would therefore expect $H(t_i;\boldsymbol{\theta})$ to represent a random sample from an exponential one distribution. With $\boldsymbol{\theta}$ estimated, $H(t_i;\hat{\boldsymbol{\theta}})$ will behave like random sample from an exponential distribution asymptotically. Therefore, a plot of $H(t_i;\hat{\boldsymbol{\theta}})$ versus the expected order statistics from a unit exponential distribution should have approximately a straight line when the model is adequate. Alternatively, one can treat the residuals as a set of possibly censored observations and derive their Kaplan-Meier estimates $\hat{S}[H(t_i;\hat{\boldsymbol{\theta}})]$. Since $-\log S(t) = H(t)$ (see Equation (2.7) and (2.9)), a plot of $-\log\{\hat{S}[H(t_i;\hat{\boldsymbol{\theta}})]\}$ versus $H(t_i;\hat{\boldsymbol{\theta}})$ should be roughly a straight line with unit slope when the original model is adequate. Note that this technique is often considered as an informal graphical method to provide insight, and are mainly useful for parametric models (Cook and Lawless 2007).

2.7.2 Proportionality Assumption for PH Models

One of the key assumptions of the PH models is that the hazard ratio comparing any two specifications of the covariates is constant over time, commonly know as the PH assumption. In other words, a PH model assumes that each covariate has a multiplicative effect on the hazard function that is constant over time. If hazards are not proportional, it indicates that the linear component of the model varies with time. This assumption is of substantial importance to check the adequacy of a PH model. There are several methods available to check the validity of the PH assumption, including a graphical technique that considers empirical plots of the log-log survival curves, and a formal statistical test by fitting an expanded model with time-dependent covariates. These techniques are particularly useful for parametric PH models.

Graphical approaches are visual aids for non-proportionality, which can provide insight into the temporality and the extent of non-proportionality that is otherwise difficult to obtain using statistical methods. Conversely, graphical methods involve a moderate degree of subjectivity in interpretation. Statistical tests typically inspect for the lack of fit of a PH model by adding time-dependent covariates. Since the Cox model is semi-parametric with the baseline hazard function

completely unspecified, special techniques are developed to check the PH assumption for the Cox model. Among these, a graphical procedure based on the Schoenfeld residuals (Schoenfeld 1982) is implemented in many statistical software packages. All these techniques are briefly described below.

Graphical Approach

Taking logs of the survivor function (2.24) twice, we get

$$\log[-\log S(t;\mathbf{z})] = \mathbf{z}'\boldsymbol{\beta} + \log[-\log S_0(t)]. \tag{2.33}$$

For two different specifications of the covariate vector corresponding to two different individuals, say $\mathbf{z}$ and $\mathbf{z}^*$, we then have

$$\log[-\log S(t;\mathbf{z})] - \log[-\log S(t;\mathbf{z}^*)] = (\mathbf{z}-\mathbf{z}^*)'\boldsymbol{\beta}, \tag{2.34}$$

which is independent of time t. Thus, a plot of the estimated log-log survival curves for two individuals on the same graph should be approximately parallel if the PH assumption is satisfied. A simple approach is to consider each covariate at a time, and check the PH assumption for a covariate by plotting log-log survival curves based on Kaplan-Meier estimates. For example, if the covariate is a binary predictor, then the log-log Kaplan-Meier curves for the two categories should be approximately parallel if the PH assumption is satisfied. To assess the PH assumption for a continuous covariate using this approach, the continuous covariate must be categorized into, say, low, medium and high.

There are three problems associated with this graphical approach (Kleinbaum and Klein 2012): (a) the decision about parallelism can be subjective, (b) it is a conservative method, assumes that the PH assumption is satisfied unless there is a strong evidence against it, and (c) it is not clear how to categorize a continuous covariate. Therefore, it is often desirable to use a formal statistical test

to evaluate the PH assumption as described below.

The Extended Model Approach using Time-Dependent Covariates

Since the PH assumption refers to the fact that the covariates are independent of time, a natural approach to check the PH assumption is to incorporate time-dependent covariates into the model and then test whether these covariates are statistically significant. For example, to check the PH assumption for a covariate z, we can take into account an additional time-dependent covariate of the form $z \times g(t)$, where $g(t)$ is a known function of t. Then, a test of the regression coefficient associated with $z \times g(t)$ reveals whether the covariate z can assumed to be independent of time. More generally, if there are p covariates $\mathbf{z} = (z_1, z_2, \ldots, z_p)'$ with associated regression coefficients $\boldsymbol{\beta} = (\beta_1, \beta_2, \ldots, \beta_p)'$, then the expanded model can be written as

$$h(t; \mathbf{z}) = h_0(t) e^{\mathbf{z}'\boldsymbol{\beta} + \beta_1^* z_1 g(t) + \beta_2^* z_2 g(t) + \ldots + \beta_p^* z_p g(t)}. \tag{2.35}$$

A test of $\mathrm{H}_0 : \beta_j^* = 0$ tests whether z_j affects the hazard function multiplicatively (i.e., whether the PH assumption is satisfied for z_j). This test can be carried out under the assumption of the asymptotic normality of $\hat{\beta}_j^*$: $\hat{\beta}_j^* / \mathrm{se}(\hat{\beta}_j^*) \sim N(0, 1)$ under H_0. A Wald test can be used to test $\mathrm{H}_0 : \beta_1^* = \beta_2^* = \ldots = \beta_p^* = 0$, which is a global test for the model as a whole. Note that the Wald test is based on $\boldsymbol{\beta}^{*\prime}[\mathrm{cov}(\boldsymbol{\beta}^*)]^{-1}\boldsymbol{\beta}^* \sim \chi_p^2$ under H_0, where $\boldsymbol{\beta}^* = (\beta_1^*, \beta_2^*, \ldots, \beta_p^*)'$.

The function $g(t)$ is commonly assumed to have a simple form. For example, $g(t)$ can be defined as an indicator function $I(t \geq t_0)$, which equals 1 if t is greater than or equal to a pre-specified value t_0 and 0 otherwise (Kleinbaum and Klein 2012). Another widely used approach is to assume a continuous function $g(t) = \log(t)$ or simply $g(t) = t$ (Lawless 2003).

Schoennfeld Residuals for the Cox PH Model

The hazard-based residuals as described above are heavily dependent on the observed survival times, and require an estimate of the cumulative hazard function (Collett 2003). To overcome

these problems, Schoenfeld (1982) proposed residuals based on the score functions of the partial likelihood function. Subsequently, methods based on Schoenfeld residuals were developed to check the PH assumption in the Cox PH model (Grambsch and Therneau 1994). The description of these residuals (see below) is taken from Collett (2003) and Lawless (2003). Note that the Schoenfeld residuals are defined for each covariate separately.

The i^{th} Schoenfeld residual for the j^{th} covariate z_j is defined by

$$\hat{S}_{ij} = \delta_i(z_{ij} - a_{ij}), \tag{2.36}$$

where

$$a_{ij} = \frac{\sum_{l \in R(t_i)} z_{lj} \exp(\mathbf{z}_l' \hat{\boldsymbol{\beta}})}{\sum_{l \in R(t_i)} \exp(\mathbf{z}_l' \hat{\boldsymbol{\beta}})}, \tag{2.37}$$

and $\sum_{i=1}^{n} \hat{S}_{ij}$ is an estimate of the first derivative of the partial log-likelihood function (2.31) with respect to β_j, $j = 1, 2, \ldots, p$ (i.e., the j^{th} component of the score vector evaluated at $\hat{\boldsymbol{\beta}}$). Using the properties of the score function, we have (a) Schoenfeld residuals for a covariate must sum to zero, (b) the expected value of $\hat{S}_{ij}$ is zero in large samples, and (c) Schoenfeld residuals are uncorrelated with one another. Grambsch and Therneau (1994) showed that if the expanded model (2.35) is correct, but the model

$$h(t; \mathbf{z}) = h_0(t) e^{\mathbf{z}'\boldsymbol{\beta}} \tag{2.38}$$

is fitted (i.e., a model with constant regression coefficients), then

$$E(\hat{S}_{ij}^{*} + \hat{\beta}_j) = \beta_j + \beta_j^{*} g(t) \tag{2.39}$$

for $j = 1, 2, \ldots, p$, where $\hat{S}_{ij}^{*}$ is the j^{th} element of $V(t_i, \hat{\boldsymbol{\beta}})^{-1} \hat{\mathbf{S}}_i$, with $\hat{\mathbf{S}}_i = (\hat{S}_{i1}, \hat{S}_{i2}, \ldots, \hat{S}_{ip})$,

$$V(t_i, \boldsymbol{\beta}) = \frac{\sum_{l \in R(t_i)} \exp(\mathbf{z}_l' \boldsymbol{\beta})(\mathbf{z}_l - \mathbf{a}_i)(\mathbf{z}_l - \mathbf{a}_i)'}{\sum_{l \in R(t_i)} \exp(\mathbf{z}_l' \boldsymbol{\beta})}, \tag{2.40}$$

and $\mathbf{a}_i = (a_{i1}, a_{i2}, \ldots, a_{ip})'$. Thus, if the proportionality assumption is satisfied for the covariate

z_j ($j = 1, 2, \ldots, p$), a smooth curve through the points of the plot of $\hat{S}^*_{ij}$ versus $g(t_i)$ should be approximately horizontal at level $\hat{\beta}_j$; a trend in the plot suggests a time-dependent effect of the covariate. Grambsch and Therneau (1994) also developed methods to test $\beta^*_j = 0$ and $\beta^*_1 = \beta^*_2 = \ldots = \beta^*_p = 0$ based on the residuals $\hat{S}^*_{ij}$. These methods are implemented in some software packages, including R (R Core Team 2016).

2.7.3 Model Comparison

Sometimes two or more models may produce satisfactory fits to observed data. In such cases, it is desirable to compare the fits in order to assign some sort of preference to the alternatives. Although a graphical approach such as comparison of residual plots can provide insight into what is going on, it may involve subjective decision-making when it comes to preferring one model over the others. This is especially true when the fits are fairly close. Therefore, a more objective approach (e.g., statistical tests, goodness of fit criteria) is desirable for model comparison. There are many model selection criteria, derived based on a variety of principles such as minimizing information loss (Akaike 1974), maximizing posterior probability (Schwarz et al. 1978), deviance information criterion (Spiegelhalter et al. 2002) and testing nested models. Below in this section, we present the likelihood ratio method to test nested models and the Akaike information criterion to compare model fits that are not necessarily nested.

Comparing Nested Models

Consider the models $M_1(\boldsymbol{\theta})$ and $M_2(\boldsymbol{\theta}^*)$, where $\boldsymbol{\theta}$ and $\boldsymbol{\theta}^*$ are the parameter vectors for the two models, respectively, with $\boldsymbol{\theta}$ is a subset of $\boldsymbol{\theta}^*$. Then, $M_1(\boldsymbol{\theta})$ is said to be parametrically nested within $M_2(\boldsymbol{\theta}^*)$ (Collett 2003). For example, the Weibull distribution is characterized by a rate parameter ρ and a shape parameter κ, whereas the exponentiated Weibull model is characterized by a rate parameter ρ and two shape parameters κ and γ. Since the Weibull distribution is a special case of the exponentiated Weibull distribution with $\gamma = 1$ (see Section 2.4.7), the Weibull

distribution is nested within the exponentiated Weibull model. The likelihood ratio test can be used to test the goodness of fit of $M_1(\boldsymbol{\theta})$ as a submodel of $M_2(\boldsymbol{\theta}^*)$. Let $\hat{\ell}_1(\hat{\boldsymbol{\theta}})$ and $\hat{\ell}_2(\hat{\boldsymbol{\theta}}^*)$ be the maximized log-likelihood functions for $M_1(\boldsymbol{\theta})$ and $M_2(\boldsymbol{\theta}^*)$, respectively. Under the null hypothesis H_0 : $M_1(\boldsymbol{\theta})$ fits the data well, the statistic $\Lambda = -2(\hat{\ell}_1(\hat{\boldsymbol{\theta}}) - \hat{\ell}_2(\hat{\boldsymbol{\theta}}^*))$ has an asymptotic chi-squared distribution with degrees of freedom equal to the difference between the number of parameters being estimated under the two models. If the null hypothesis is rejected, then $M_2(\boldsymbol{\theta}^*)$ is preferred over $M_1(\boldsymbol{\theta})$ in terms of adequacy of the model in describing the data.

Akaike Information Criterion

The Akaike information criterion (AIC) (Akaike 1974) is defined as

$$\text{AIC} = -2\log(\text{maximized likelihood}) + 2(\text{number of estimated parameters in the model}), \quad (2.41)$$

where $D = -2\log(\text{maximized likelihood})$ is called deviance. The deviance represents the degree of inaccuracy when the maximum likelihood estimates are used, and 2(number of estimated parameters in the model) is considered as a penalizing factor associated with the complexity of a model. When comparing two or more models, we prefer the one with the lowest AIC value. A rule of thumb is that if $\text{AIC}_\text{M} - \text{AIC}_\text{min} > 2$, then there is considerably less support for Model M compared to the model with minimum AIC (Burnham and Anderson 2002). Note that AIC can be used to compare models (in terms of the best short-term predictions) that are not necessarily nested.

2.8 Recurrent Event Modeling

The analysis of survival data involves modeling the time to the occurrence of an event of interest. In many biomedical studies and engineering, the event may not necessarily be fatal. Thus, an individual can experience the event repeatedly over time. Examples include ipsilateral breast tumor

recurrences after conserving surgery, repeated hospital admissions due to asthma, falls in elderly patients and upper respiratory infections. Such processes are called recurrent event processes, and the data generated by such processes are called recurrent event data. The standard survival models as described the previous sections cannot directly handle an analysis of such data. Statistical methods based on counting processes and intensity functions play the canonical role to model and analyze recurrent event data (Andersen and Gill 1982). There are several text books emphasizing the mathematical details of counting process and intensity functions (e.g., Andersen et al. 2012, Therneau and Grambsch 2013, Cook and Lawless 2007). In this section, we present the basic idea underlying the recurrent event processes, and the statistical models commonly used to analyze such data.

2.8.1 Modeling Framework and Data Structure

The canonical framework for the counting process formulation is the Poisson process. For a Poisson process, events occur randomly in such a way that the numbers of events in non-overlapping time intervals are statistically independent. Under these assumptions, it is possible to generalize the survival analysis for single event to recurrent event analysis. Consequently, the emphasis changes from modeling the hazard of a survival function to modeling the intensity or rate of a point process. Now we give a brief review of the mathematical notations to describe the general framework for recurrent event data analysis under point Poisson process (Therneau and Grambsch 2013).

Consider each individual experience multiple events of the same type. In this setting, individuals are indexed by i, $i = 1, 2, \ldots, n$, and each individual's events are indexed by j, $j = 1, 2, \ldots, n_i$. For each individual i, let T_{ij} denote the time of the individual's jth event measured from a starting point, such as study start. The time an individual is observed for the events is completed by a censoring time, C_i. Let $\mathbf{z}_i$ be a $p \times 1$ vector of covariates for individual i, assumed to be fixed in time and $I(.)$ be the indicator function. Thus, the predictable processes $Y_{ij}(t)$ define when an individual is at risk of an event. We model $Y_{ij}(t) = I(\{t \leq C_i\} \text{ and } \{t \in [T_{ij-1}, T_{ij}]\})$, i.e., an individual is at

risk until experiencing the jth event immediately after the $(j-1)th$ event and remains at risk until experiencing the jth event. Further detail about the at-risk process can be found in Cook and Lawless (2007). By using the at risk process, we can write the observed part of the counting process as an intensity process. In the following section, we present the most commonly used model for recurrent event data analysis, namely, the Andersen-Gill model.

2.8.2 Andersen Gill Model

Andersen and Gill (1982) model, also referred as the proportionality intensity model is the generalization of the Cox proportional hazard model on the basis of the theory of counting processes for survival analysis. The multiplicative hazard function $h(t;\mathbf{z})$ for the i^{th} individual can be expressed as

$$h(t;\mathbf{z}_i) = Y_i(t)h_0(t)e^{\mathbf{z}_i'\boldsymbol{\beta}}, \tag{2.42}$$

where $Y_i(t)$, an indicator , equals one when the i^{th} individual is under observation (at risk) at time t and $h_0(t)$ is an unspecified hazard function. The partial likelihood for n independent individual is given as

$$L(\boldsymbol{\theta}) = \prod_{i=1}^{n}\prod_{t\geq 0}\left\{\frac{Y_i(t)\exp\{\mathbf{z}_i'(t)\boldsymbol{\beta}\}}{\sum_{j=1}^{n}Y_j(t)\exp\{\mathbf{z}_i'(t)\boldsymbol{\beta}\}}\right\}^{\delta_i(t)}, \tag{2.43}$$

where $\delta_i(t) = 1$ if the i^{th} person has an event at t, and 0 otherwise. Further mathematical proves and detailed description have been discussed in various references such as Andersen and Gill (1982) and Therneau et al. (1990).

The basic formulation of the Andersen-Gill model is similar to that of the Cox PH model, with the baseline intensity function assumed arbitrary. Note that the baseline intensity function can also be specified parametrically, leading to a fully parametric model. In summary, the general concept of analyzing recurrent event data by point process approach is to model intensity in such a way to make inference on the effect of covariates on the occurrence of recurrent events without losing

much flexibility and information in the baseline intensity (Therneau and Grambsch 2013).

2.9 Joint Modeling

In many clinical or biomedical studies, a longitudinal response is observed along with an observation of the time to the occurrence of an event; the event can be timed from the beginning of an observation period, resulting in survival or time-to-event data. A typical goal in such studies is to investigate the effects of the longitudinal response on the development of the event. Moreover, it is also of particular interest to understand the within-individual trends of the longitudinal response. Formalizing these objectives is straightforward conceptually, but addressing them in practice is quite challenging by the nature of the data actually observed. Valid inference from such data sets requires a comprehensive framework to quantify the underlying relationship between the time-to-event process and the longitudinal reponse.

There are many well-established methods in statistical literature to analyze these two processes separately. For example, linear mixed effects (LME) models based on maximum and restricted likelihood (Laird and Ware 1982) and marginal and transitional models based on generalized estimating equations (GEE) approach (Liang and Zeger 1986) have been extensively used in longitudinal studies. On the other hand, PH and AFT models are widely used to analyze time-to-event data. If the longitudinal response is an internal covariate of the event process and the primary interest is to explore its effects on the time to the occurrence of an event, separate analyses are not statistically efficient. This is because a standard survival model does not take into account various features of a longitudinal response (e.g., missing data, measurement error) in determining its effects on the time-to-event process. A more efficient way to study these two processes is to assume that they are related through some unobserved latent effects (i.e., random effects). Such unobservable variables are time-dependent measurements taken on individuals under study. As an example, CD4 cell counts for HIV-infected patients (longitudinal response) is an internal covariate

for the time-to-event process involving times until death. Note that CD4 cell counts are subject to measurement errors and missing values (data are collected only intermittently over time). Separate analyses of these two processes do not explicitly acknowledge the general features of the internal covariate (longitudinal response) in studying its effects on survival times, thereby may lead to biased estimates of the regression parameters (Tsiatis and Davidian 2004). Thus, the modern approach to analyze these types of data is to jointly model the survival outcome and the longitudinal response through shared random effect(s), which allows effective quantification of the association between repeated measures and event times (Wulfsohn and Tsiatis 1997, Henderson et al. 2000, Rizopoulos 2012).

A typical joint model setting is to assume a mixed-effects model for the longitudinal data (Fitzmaurice et al. 2011) and a proportional hazards model for the survival data, with the two models sharing some random effects. An overview of early work on joint models can be found in many articles, including Tsiatis and Davidian (2004), Wulfsohn and Tsiatis (1997) and Wang and Taylor (2001). A detailed description of joint model formulation is given in Chapter 5.

For valid statistical inference in joint modeling paradigm, there are two main estimation procedures: two-stage method and the maximum likelihood estimation (Self and Pawitan 1992, Tsiatis et al. 1995, Henderson et al. 2000). The likelihood method is usually preferred because of the well-known features of the maximum likelihood estimators, including asymptotic unbiasedness, consistency, efficiency and the invariance property. However, the likelihood method for joint models is highly complicated, involving evaluation of multiple integrals with respect to time for the event process and random effects for the longitudinal process (Rizopoulos 2012). Although the integral with respect to time can be well approximated using the Gauss-Kronrod rule (Press et al. 2007), the integrals with respect to the random effects are computationally gridlock, especially when the dimensionality of the random effects increases. Bayesian estimation for joint models has also been considered (Xu and Zeger 2001, Guo and Carlin 2004). Both Bayesian and likelihood procedures rely on specification of an appropriate likelihood for the joint model parameters. However, it may

be advantageous to use a Bayesian approach because (a) integrations with respect to the random effects can be avoided by using the Markov Chain Monte Carlo (MCMC) algorithm, (b) asymptotic approximations for statistical inference are not necessary, (c) model assessment is relatively more straightforward, (d) computational implementation is much easier, and (d) prior information can be incorporated into the inference procedure. With this motivation, we propose a Bayesian approach for joint models. Note that availability of generic statistical software for Bayesian inference (e.g., WinBUGS, JAGS, STAN) makes Bayesian implementation of a complicated model fairly simpler. In the following section, we provide a brief review of Bayesian inference in general.

2.10 Bayesian Inference

In the Bayesian inferential paradigm, probability distributions are associated with the parameters of the likelihood, as if the parameters were random variables. The main idea is to combine data and prior knowledge on a parameter (or a vector of parameters) to determine its posterior distribution (the conditional density of the parameter given the data). The prior knowledge is supplied in the form of a prior distribution, which quantifies information and uncertainty about the parameter prior to any data being gathered (Gelman et al. 2013).

Bayesian inference is based on Monte Carlo samples (MCMC) drawn from the posterior distribution using an MCMC algorithm such as the Gibbs sampler. Markov chain Monte Carlo (MCMC) is essentially Monte Carlo integration using Markov chains. Bayesians, and sometimes also frequentists, need to integrate over possibly high-dimensional probability distributions to make inference about model parameters or to make predictions. Monte Carlo integration draws samples from the required distribution, and then forms sample averages to approximate expectations. Markov chain Monte Carlo draws these samples by running a cleverly constructed Markov chain for a long time. Monte Carlo sampling can be done using publicly available WinBUGS software (Lunn et al. 2000). In the following section, we will not attempt to give an exhaustive description of conver-

gence techniques to assess performance of MCMC, but rather highlight some of the key ideas (for a detail description, see, Riggelsen 2008, Gelman et al. 2013) .

2.10.1 Monitoring MCMC Convergence

Whenever one runs MCMC, it is important to assess its performance. However, it is often difficult to decide at what point it would be reasonable to believe that the samples are accurately approximate the underlying stationary distribution of the Markov chain. In order to assess whether a Markov chain converges to its stationary distribution, some diagnostic tools are usually considered, including mixing, burn-in and run-length. The mixing property of a chain includes how quickly a chain "forgets" its initial values and how quickly it fully explores the support and the shape of the target distribution. When a chain at some point reaches around the mode of the target distribution, it is possible that it may stay there forever. If this is the case, even though the chain does not fully explore the support and the shape of the target distribution, a convergence diagnostic may indicate that the chain has converged to the stationary distribution. A partial solution to this problem is to run several independent chains, and then investigate the within-chain and between-chain behavior. Moreover, the dependence of a chain on the starting value may remain strong even after running the chain for a sufficiently long time. If a chain starts with an initial value that is far from the posterior mode, this dependence may make the chain converge slowly. To reduce the severity of this problem, an initial iterations are discarded as a burn-in period from a chain. A chain that has poor mixing properties generally exhibits slow decay of autocorrelation. Therefore, it is good practice for the inference to be based on every l^{th} iteration of a chain, with l set to some value high enough that successive draws are approximately independent. This strategy is known as thinning.

Trace and density plots are also very useful graphical tools for visualizing the convergence of Markov chains. Note that a density plot is a smoothed histogram of the MCMC samples used to approximate the posterior density, whereas a trace plot is plot of observed values of MCMC

samples vs MCMC iterations. A trace plot is useful to assess convergence and mixing of a Markov chain; a clear trend in the trace plot indicates that stationarity has not been achieved. This in turn suggests that a longer run is necessary. A chain that is mixing well will quickly move away from its starting value, no matter where it started, and the samples will wiggle about vigorously in the region supported by the posterior density. Another way to check for convergence is to look at the autocorrelations between the samples generated by MCMC. Autocorrelation plot displays the serial correlation in the chain at different lags of iteration. For a highly autocorrelated chain, the sampler is slow to explore the entire support of the posterior distribution. Typically, the autocorrelation should become smaller as the lag increases. If this is not the case, thinning should be increased. The Gelman-Rubin statistic R (Gelman and Rubin 1992) is useful diagnostic measure. The technique is based on a comparison of within-chain and between-chain variances. Values of R substantially above 1 indicate lack of convergence. Some authors suggest that $R < 1.2$ is acceptable (Brooks and Gelman 1998).

There are many model selection criteria in Bayesian analysis, derived based on a variety of principles. These include deviance information criterion (DIC) (Spiegelhalter et al. 2002), BIC or the Schwarz criterion (Schwarz et al. 1978), and Watanabe-Akaike information criterion (WAIC) (Watanabe 2010). Perhaps, DIC is the most widely used criterion for model comparison in Bayesian analysis. It is derived based on two principles: (i) goodness of fit measured via the deviance statistic, and (ii) model complexity measured by an estimate of the effective number of parameters, denoted by p_D. When comparing two or more models, it is suggested that $\text{DIC}_\text{M} - \text{DIC}_\text{min} > 10$ or if the difference lies between 5 and 10, then there is considerably less support for Model M compared to the model with minimum DIC. However, $\text{DIC}_\text{M} - \text{DIC}_\text{min} < 5$ show no support for a model with the lowest DIC and may lead to misleading inference. (https://www.mrc-bsu.cam.ac.uk/software/bugs/the-bugs-project-dic/#q8). Note that like any other statistical tools, DIC suffers some drawbacks. In particular, there are some criticisms towards the measure of model complexity: (a) p_D is not invariant to reparametrization, (b) it is not

based on a proper predictive criterion and has a weak theoretical justification, and (c) p_D can be negative when there is substantial conflict between prior and data, or when the posterior distribution for a parameter is extremely asymmetric or bimodal; see Spiegelhalter et al. (2014). Nevertheless, DIC has several desirable properties, including the fact that it can be calculated when noninformative or improper priors are used. Moreover, it is simple to calculate using MCMC simulation and is routinely implemented in the WinBUGS software package. On the other hand, although WAIC is preferred by many Statisticians, it is computationally expensive. In this thesis, we consider DIC for model comparison (our future plan is to consider/implement WAIC).

2.11 Conclusion

In this chapter, we have presented some of the key concepts of modeling time-to-event-data. As mentioned in Chapter 1, we propose a generalization of the log-logistic distribution and develop regression methodology based on this model. The published manuscript out of this work is presented in Chapter 3.

CHAPTER 3

THE GENERALIZED LOG-LOGISTIC MODEL

In this chapter, we present our published manuscript[1] about generalization of the log-logistic distribution in proportional hazard framework. Note that the materials of this chapter have been reproduced from our article Khan and Khosa (2016). Proportional hazard (PH) models can be formulated with or without assuming a probability distribution for survival times. The former assumption leads to parametric models, whereas the latter leads to the semi-parametric Cox model which is by far the most popular in survival analysis. However, a parametric model may lead to more efficient estimates than the Cox model under certain conditions (Hjort 1992). Only a few parametric models are closed under the PH assumption, the most common of which is the Weibull that accommodates only monotone hazard functions. We propose a generalization of the log-logistic distribution that belongs to the PH family. It has properties similar to those of log-logistic, and approaches the Weibull in the limit. These features enable it to handle both monotone and nonmonotone hazard functions. Application to four data sets and a simulation study revealed that the model could potentially be very useful in describing different types of time-to-event data.

The rest of this chapter is organized as follow: Section 3.1 outlines the brief introduction and motivation of the study. In Section 3.2, we have introduced the generalized log-logistic model and discussed estimation and testing of the parameters using the maximum likelihood method. The proposed method is then illustrated with applications to four data sets, one of which involves joint modeling of time-to-event and longitudinal data (Section 3.3). In Section 3.4, a simulation

[1]Khan, S. A. and Khosa, S. K. (2016). Generalized log-logistic proportional hazard model with applications in survival analysis. Journal of Statistical Distributions and Applications, 3(1):16

study is presented to evaluate the performance of generalized log-logistic in comparison with other commonly used PH models to describe different types of time-to-event data. We conclude in Section 3.5 by summarizing our findings.

3.1 Introduction

Proportional hazard (PH) models play a vital role in analyzing time-to-event data. One of the appealing features of PH models is that the regression coefficients have relative risk interpretation, which is preferred by many Clinicians. The Cox PH model (Cox 1972) is the most popular in survival analysis mainly because of two reasons: (a) no assumption is required about the probability distribution of survival times (i.e., a semi-parametric model), and (b) it usually fits the data well no matter which parametric model is appropriate. In contrast, distributional assumption is required for a fully parametric PH model (Kalbfleisch 2002, Lawless 2003). This also leads to the added requirement of checking the appropriateness of the chosen distribution. Nevertheless, as demonstrated by Efron (1977) and Oakes (1977), parametric models lead to more efficient estimates than Cox's model under certain conditions. More specifically, if the distributional assumption is valid, a parametric model leads to smaller standard errors of the estimates than would be in the absence of a distributional assumption (Collett 2003). Moreover, the use of Cox PH in joint modeling of time-to-event and longitudinal data (Wulfsohn and Tsiatis 1997) usually leads to an underestimation of the standard errors of the parameter estimates (Hsieh et al. 2006, Rizopoulos 2012), and therefore most methods for joint modeling are based on parametric response distributions. Regarding the choice between a parametric and Cox's PH model, Nardi and Schemper (2003) suggested to use a richer parametric model or simply the Cox's model in case of an unsatisfactory fit of the chosen probability distribution.

The most commonly used parametric time-to-event models are the Weibull, log-logistic and log-normal distributions. The log-logistic and log-normal distributions belong to the accelerated

failure time (AFT) family, and are useful in modeling nonmonotone hazard rates (Lawless 2003). Note that the log-logistic also accommodates decreasing hazard functions. Only a few parametric models are closed under PH assumption, the most common of which is the Weibull that accommodates only monotone hazard functions. In fact, Weibull is the only distribution that is closed under both AFT and PH families (Kalbfleisch 2002). Mudholkar et al. (1996) proposed a generalization of the Weibull distribution which permits parametric PH regression modelling. It is a three-parameter distribution and is capable of modeling both monotone and nonmonotone hazard functions. One difficulty with this model is that it is nonregular (the support depends on some parameters) in the case of increasing hazard functions, and therefore the standard maximum likelihood asymptotics do not hold. In this paper, we propose a simple extension of the log-logistic model which is closed under the PH relationship. The proposed generalized log-logistic model is a three-parameter distribution, and has characteristics similar to those of the log-logistic model. Moreover, it approaches to Weibull in the limit. These features enable it to satisfactorily handle both monotone and nonmonotone (unimodal) hazard functions.

3.2 The Generalized Log-logistic Model

The generalized log-logistic distribution for a nonnegative random variable T can be conveniently specified in terms of the hazard function as follows:

$$h(t;\boldsymbol{\alpha}) = \frac{\kappa\rho(\rho t)^{\kappa-1}}{1+(\gamma t)^{\kappa}}, \qquad t > 0, \tag{3.1}$$

where $\rho > 0$, $\kappa > 0$ and $\gamma > 0$ are parameters and $\boldsymbol{\alpha} = (\kappa, \gamma, \rho)'$. If γ depends on ρ via $\gamma = \rho$ and $\gamma = \rho\eta^{-1/\kappa}$ with $\eta > 0$, then (3.1) reduces to the hazard function of the log-logistic (Lawless 2003) and Burr XII (Wang et al. 2008) distributions, respectively. Taking γ not dependent on ρ, it is easy to verify that (3.1) is closed under PH relationship. The hazard function is monotone decreasing when $\kappa \leq 1$, and unimodal when $\kappa > 1$ (i.e., $h(0) = 0$, increases to a maximum at $[(\kappa-1)/\gamma^{\kappa}]^{1/\kappa}$,

and then approaches zero monotonically as $t \to \infty$). Note that (3.1) approaches to the Weibull hazard function as $\gamma^\kappa \to 0$. This particular feature of the generalized log-logistic model enables it to handle monotone increasing hazard satisfactorily via $\kappa > 1$ and γ small (close to zero).

The survivor function, probability density function and cumulative hazard function of the generalized log-logistic distribution are given as,

$$S(t;\boldsymbol{\alpha}) = [1 + (\gamma t)^\kappa]^{-\frac{\rho^\kappa}{\gamma^\kappa}}, \tag{3.2}$$

$$f(t;\boldsymbol{\alpha}) = \frac{\kappa\rho(\rho t)^{\kappa-1}}{[1 + (\gamma t)^\kappa]^{\frac{\rho^\kappa}{\gamma^\kappa}+1}}, \tag{3.3}$$

$$H(t;\boldsymbol{\alpha}) = \frac{\rho^\kappa}{\gamma^\kappa}\log[1 + (\gamma t)^\kappa]. \tag{3.4}$$

The median of the distribution is $\frac{\left(2^{\frac{\gamma^\kappa}{\rho^\kappa}}-1\right)^{\frac{1}{\kappa}}}{\gamma}$, and the r^{th} moment is

$$E(T^r) = \frac{\rho^\kappa}{\gamma^{\kappa+r}}\,\frac{\Gamma(\frac{\rho^\kappa}{\gamma^\kappa} - \frac{r}{\kappa})\Gamma(\frac{r}{\kappa}+1)}{\Gamma(\frac{\rho^\kappa}{\gamma^\kappa}+1)} \quad \text{provided} \quad \frac{\kappa\rho^\kappa}{\gamma^\kappa} > r.$$

In particular, the mean is $E(T) = \frac{\rho^\kappa}{\gamma^\kappa}\frac{\Gamma(\frac{\rho^\kappa}{\gamma^\kappa}-\frac{1}{\kappa})\Gamma(\frac{1}{\kappa}+1)}{\Gamma(\frac{\rho^\kappa}{\gamma^\kappa}+1)}$ provided $\frac{\kappa\rho^\kappa}{\gamma^\kappa} > 1$.

The generalized log-logistic PH with covariates can be expressed as

$$h(t;\mathbf{z}) = h_0(t;\boldsymbol{\alpha})\, e^{\mathbf{z}'\boldsymbol{\beta}} \tag{3.5}$$

where $h_0(t;\boldsymbol{\alpha})$ is the baseline hazard function given by (3.1), $\mathbf{z} = (z_1, z_2, \ldots, z_p)'$ is the vector of covariates and $\boldsymbol{\beta} = (\beta_1, \beta_2, \ldots, \beta_p)'$ is the corresponding vector of regression parameters. Note that (3.5) reduces to the Weibull and Cox PH models if $h_0(t;\boldsymbol{\alpha}) = \kappa\rho(\rho t)^\kappa$ and $h_0(t;\boldsymbol{\alpha}) = h_0(t)$, respectively, where $h_0(t)$ is an arbitrary unspecified baseline hazard function.

3.2.1 Estimation

Suppose that a censored random sample consisting of data $(t_i, \delta_i, \mathbf{z}_i)$, $i = 1, 2, \ldots, n$, is available, where t_i is a lifetime or censoring time according to whether $\delta_i = 1$ or 0, respectively, and $\mathbf{z}_i =$

$(z_{i1}, z_{i2}, \ldots, z_{ip})'$ is the vector of covariates for the i^{th} individual. Letting $m = \sum_{i=1}^{n} \delta_i$, $a_i = \exp(\mathbf{z}_i'\boldsymbol{\beta})$ and $b_i = (\gamma t_i)^{\kappa}$, the log-likelihood function for the generalized log-logistic PH can be written as

$$\ell(\boldsymbol{\theta}) = m\log\kappa + m\kappa\log\rho + (\kappa-1)\sum_{i=1}^{n}\delta_i\log t_i - \sum_{i=1}^{n}\delta_i\log(1+b_i) + \sum_{i=1}^{n}\delta_i\log a_i - \left(\frac{\rho}{\gamma}\right)^{\kappa}\sum_{i=1}^{n}a_i\log(1+b_i), \tag{3.6}$$

where $\boldsymbol{\theta} = (\boldsymbol{\alpha}', \boldsymbol{\beta}')'$. The first derivatives of the log-likelihood function are

$$\frac{\partial\ell(\boldsymbol{\theta})}{\partial\kappa} = \frac{m}{\kappa} + m\log\rho + \sum_{i=1}^{n}\delta_i\log t_i - \frac{1}{\kappa}\sum_{i=1}^{n}\delta_i b_i c_i - \left(\frac{\rho}{\gamma}\right)^{\kappa}\left(\frac{1}{\kappa}\right)\sum_{i=1}^{n}a_i b_i c_i - \left(\frac{\rho}{\gamma}\right)^{\kappa}\log\left(\frac{\rho}{\gamma}\right)\sum_{i=1}^{n}a_i\log(1+b_i), \tag{3.7}$$

$$\frac{\partial\ell(\boldsymbol{\theta})}{\partial\gamma} = -\left(\frac{\kappa}{\gamma}\right)\sum_{i=1}^{n}\delta_i d_i - \left(\frac{\kappa}{\gamma}\right)\left(\frac{\rho}{\gamma}\right)^{\kappa}\sum_{i=1}^{n}a_i d_i - \left(\frac{\kappa}{\gamma}\right)\left(\frac{\rho}{\gamma}\right)^{\kappa}\sum_{i=1}^{n}a_i\log(1-d_i), \tag{3.8}$$

$$\frac{\partial\ell(\boldsymbol{\theta})}{\partial\rho} = \frac{m\kappa}{\rho} - \left(\frac{\kappa}{\rho}\right)\left(\frac{\rho}{\gamma}\right)^{\kappa}\sum_{i=1}^{n}a_i\log(1+b_i), \tag{3.9}$$

$$\frac{\partial\ell(\boldsymbol{\theta})}{\partial\beta_j} = \sum_{i=1}^{n}\delta_i z_{ij} - \left(\frac{\rho}{\gamma}\right)^{\kappa}\sum_{i=1}^{n}a_i\log(1+b_i)z_{ij} \quad \text{for } j = 1, 2, \ldots, p, \tag{3.10}$$

where $c_i = \log b_i/(1+b_i)$ and $d_i = b_i/(1+b_i)$ (see Appendix A.1). To improve the convergence of iterative procedures for maximum likelihood estimation and the accuracy of large-sample methods, we remove range restrictions on parameters through the parameterizations $\boldsymbol{\alpha}^* = (\kappa^*, \gamma^*, \rho^*)'$, where $\kappa^* = \log\kappa$, $\gamma^* = \log\gamma$ and $\rho^* = \log\rho$. The maximum likelihood estimate of $\boldsymbol{\theta}^* = (\boldsymbol{\alpha}^{*\prime}, \boldsymbol{\beta}')'$ can then be obtained by solving the equations $\partial\ell(\boldsymbol{\theta}^*)/\partial\kappa^* = 0$, $\partial\ell(\boldsymbol{\theta}^*)/\partial\gamma^* = 0$, $\partial\ell(\boldsymbol{\theta}^*)/\partial\rho^* = 0$ and $\partial\ell(\boldsymbol{\theta}^*)/\partial\beta_j = 0$ iteratively, where (see Appendix A.1)

$$\frac{\partial\ell(\boldsymbol{\theta}^*)}{\partial\kappa^*} = \left[\kappa\left(\frac{\partial\ell(\boldsymbol{\theta})}{\partial\kappa}\right)\right]_{\boldsymbol{\alpha}=\exp(\boldsymbol{\alpha}^*)}, \quad \frac{\partial\ell(\boldsymbol{\theta}^*)}{\partial\gamma^*} = \left[\gamma\left(\frac{\partial\ell(\boldsymbol{\theta})}{\partial\gamma}\right)\right]_{\boldsymbol{\alpha}=\exp(\boldsymbol{\alpha}^*)},$$

$$\frac{\partial \ell(\boldsymbol{\theta}^*)}{\partial \rho^*} = \Big[\rho\Big(\frac{\partial \ell(\boldsymbol{\theta})}{\partial \rho}\Big)\Big]_{\alpha=\exp(\alpha^*)}, \quad \frac{\partial \ell(\boldsymbol{\theta}^*)}{\partial \beta_j} = \Big[\frac{\partial \ell(\boldsymbol{\theta})}{\partial \beta_j}\Big]_{\alpha=\exp(\alpha^*)}.$$

Many software packages have reliable optimization procedures to maximize log-likelihood functions. We wrote our computer code in R (R Core Team 2016), and used the function *nlminb* for optimization (see Appendix A.2).

3.2.2 Initial Values

We may use Weibull, log-logistic and Cox PH fits to generate initial values in solving the equations $\partial \ell(\boldsymbol{\theta}^*)/\partial \kappa^* = 0$, $\partial \ell(\boldsymbol{\theta}^*)/\partial \gamma^* = 0$, $\partial \ell(\boldsymbol{\theta}^*)/\partial \rho^* = 0$ and $\partial \ell(\boldsymbol{\theta}^*)/\partial \beta_j = 0$. Let $\hat{\kappa}_1$ and $\hat{\rho}_1$ be the maximum likelihood estimates of the Weibull shape and scale parameters, respectively, $\hat{\kappa}_2$ and $\hat{\rho}_2$ the maximum likelihood estimates of the log-logistic shape and scale parameters, respectively, and $\hat{\boldsymbol{\beta}}^*$ the estimates of the regression coefficients for the Cox PH model. Note that maximum likelihood methods for the Weibull, log-logistic and Cox PH models are available in many statistical softwares, including R (R Core Team 2016). We propose to use $\log \hat{\kappa}_1$, $\log|\hat{\kappa}_1 - \hat{\kappa}_2|$, $\log \hat{\rho}_1$ and $\hat{\boldsymbol{\beta}}^*$ as initial values for κ^*, γ^*, ρ^* and $\boldsymbol{\beta}$, respectively. If convergence is not achieved with these initial values, we propose to replace $\log \hat{\kappa}_1$ and $\log \hat{\rho}_1$ by $\log \hat{\kappa}_2$ and $\log \hat{\rho}_2$, respectively. In fitting the generalized log-logistic model to many data sets, we have not experienced any difficulty in obtaining convergence with this technique.

3.2.3 Tests and Confidence Interval

Tests and interval estimates for the model parameters are based on the approximate normality of the maximum likelihood estimators. The asymptotic distribution of $\hat{\boldsymbol{\theta}}^*$ is approximately a $(p+3)$-variate normal distribution with mean $\boldsymbol{\theta}^*$ and covariance matrix $I(\hat{\boldsymbol{\theta}}^*)^{-1}$, where

$$I(\hat{\boldsymbol{\theta}}^*) = -\begin{bmatrix} \frac{\partial^2 \ell(\boldsymbol{\theta}^*)}{\partial \kappa^{*2}} & \frac{\partial^2 \ell(\boldsymbol{\theta}^*)}{\partial \kappa^* \partial \gamma^*} & \cdots & \frac{\partial^2 \ell(\boldsymbol{\theta}^*)}{\partial \kappa^* \partial \beta_p} \\ \frac{\partial^2 \ell(\boldsymbol{\theta}^*)}{\partial \gamma^* \partial \kappa^*} & \frac{\partial^2 \ell(\boldsymbol{\theta}^*)}{\partial \gamma^{*2}} & \cdots & \frac{\partial^2 \ell(\boldsymbol{\theta}^*)}{\partial \gamma^* \partial \beta_p} \\ \vdots & \vdots & \vdots & \vdots \\ \frac{\partial^2 \ell(\boldsymbol{\theta}^*)}{\partial \beta_p \partial \kappa^*} & \frac{\partial^2 \ell(\boldsymbol{\theta}^*)}{\partial \beta_p \partial \gamma^*} & \cdots & \frac{\partial^2 \ell(\boldsymbol{\theta}^*)}{\partial \beta_p^2} \end{bmatrix}_{\boldsymbol{\theta}^* = \hat{\boldsymbol{\theta}}^*}.$$

is the $(p+3)\times(p+3)$ observed information matrix (second derivatives of $\ell(\boldsymbol{\theta}^*)$ are given Appendix A.1). Note that by the multivariate delta method, the asymptotic distribution of $\hat{\boldsymbol{\theta}}$ is also approximately normal with mean $\boldsymbol{\theta}$ and covariance matrix $D\Sigma D'$, where D is the $(p+3)\times(p+3)$ diagonal matrix $diag(\hat{\alpha}, 1, 1, \ldots, 1)$, $\Sigma = I(\hat{\boldsymbol{\theta}}^*)^{-1}$ and $\hat{\alpha} = \exp(\hat{\alpha}^*)$.

3.2.4 Generalized Log-logistic Distribution in Joint Modelling

Joint models are used to quantify association between an internal time-dependent covariate and time until an event of interest occurs (Wulfsohn and Tsiatis 1997). It involves two separate models: a model that takes into account measurement error in the time-dependent covariate to estimate its true values (longitudinal model), and another model that uses these estimated values to quantify the association between this covariate and the time to the occurrence of the event (time-to-event model). The idea behind the joint modeling technique is to couple the time-to-event model with the longitudinal model. The general framework of the maximum likelihood method and large sample theory can be found in Rizopoulos (2012). Maximization of the log-likelihood function for joint modeling is computationally challenging, as it involves evaluating multiple integrals that do not have an analytical solution, except in very special cases. The R package **JM** has been developed by Rizopoulos et al. (2010) to fit joint models using Weibull baseline hazard, piecewise-constant baseline hazard, spline approximation of the baseline hazard and unspecified baseline hazard functions. We have modified the source codes for Weibull to fit joint models using the generalized log-logistic baseline hazard function. The application of the generalized log-logistic

distribution in joint modeling is illustrated with an example in Section 3.3.4.

3.2.5 Goodness of Fit

The nonparametric estimates are useful for assessing the quality of fit of a particular parametric time-to-event model (Lawless 2003). For a model without covariate, we use the approach to simultaneously examine plots of parametric and nonparametric estimates of the survival function, superimposed on the same graph. Let $S(t;\hat{\boldsymbol{\theta}})$ and $\hat{S}(t)$ be the estimates of the survivor functions based on the parametric model of interest and the Kaplan-Meier method (Kaplan and Meier 1958), respectively. The estimates $S(t;\hat{\boldsymbol{\theta}})$ as a function of t should be close to $\hat{S}(t)$ if the parametric model is adequate. For a model with covariates, we consider residual diagnostic plots, where the residuals are defined based on the cumulative hazard function $H(t;\boldsymbol{\theta})$. If $\hat{S}(H(t;\hat{\boldsymbol{\theta}}))$ is the Kaplan-Meier estimate of $H(t;\hat{\boldsymbol{\theta}})$, then a plot of $-\log\hat{S}(H(t;\hat{\boldsymbol{\theta}}))$ versus $H(t;\hat{\boldsymbol{\theta}})$ should be roughly a straight line with unit slope when the model is adequate (Lawless 2003).

We also use the Akaike's information criterion (AIC) (Akaike 1974) to compare the fits of different models. The AIC is defined by AIC $= -2\log($maximized likelihood$)+2(p+k)$, where p is the number of covariates and k is the number of parameters of the assumed probability distribution ($k = 3$ for the generalized log-logistic model). In general, when comparing two or more models, we prefer the one with the lowest AIC value. A rule of thumb is that if $\Delta_{\mathrm{M}} = \mathrm{AIC}_{\mathrm{M}} - \mathrm{AIC}_{\mathrm{min}} > 2$, then there is considerably less support for Model M compared to the model with minimum AIC (Burnham and Anderson 2003).

3.3 Application

Three data sets are taken from the literature to demonstrate the ability of the generalized log-logistic distribution in modeling time-to-event data. The application of the generalized log-logistic PH in joint modeling is illustrated using another data set on AIDS patients. We first use the so-

called scaled Total Time on Test (TTT) of failure times to detect the shape of the hazard function (Mudholkar et al. 1996). The scaled TTT-transform and the TTT-plot were first presented in a famous paper by Barlow and Campo (1975). Let $F(t)$ denote the life distribution of certain unit (F is continuous and strictly increasing). Furthermore, let $S(t)$ be the corresponding survival function. The mean μ is calculated by integrating the survival function, i.e. $\mu = \int_0^\infty S(t)dt$. With these notations in mind, the function defined on [0,1] by $H_F^{-1}(u) = \int_0^{F^{-1}(u)} S(t)dt$ is called the scaled TTT-transform, where TTT means 'Total Time on Test'. If we have an order sample $0 = t_{(0)} \leq t_{(1)} \leq t_{(2)} \ldots \leq t_{(n)}$ of times to failure of our unit, we can get estimator of the scaled TTT-transform, called the TTT-plot. When the sample size increases to infinity the TTT-plot converges (with probability one and uniformly; see Langberg et al. (1980) for detail) to the scaled TTT-transform of the life distribution $F(t)$ from which our sample has come. The scaled TTT transform is given by $\phi(v/n) = \left[\sum_{i=1}^{v} T_{(i)} + (n-v)T_{(v)}\right] / \left(\sum_{i=1}^{n} T_{(i)}\right)$, where $T_{(i)}$ represent the order statistics of the sample, and $v = 1, 2, \ldots, n$. The hazard function is increasing, decreasing and unimodal if the plot of $(v/n, \phi(v/n))$ is concave, convex, and concave followed by convex, respectively. For the first three examples (Sections 3.3.1-3.3.3), we first fit the generalized log-logistic, Weibull and log-logistic models (without covariate) and check the appropriateness of the distributional assumption using diagnostic plots. Then, we analyze the data using regression models, and compare the fits via residual plots. Note that the regression model based on the log-logistic distribution is given by $\log T = \beta_0 + \beta_1 z_1 + \ldots + \beta_p z_p + \tau W$ where $\tau = 1/\kappa$, $\beta_0 = -\log \rho$ and W has the logistic distribution with density $f(w) = e^w/(1+e^w)^2$, $-\infty < w < \infty$. This model has an accelerated life interpretation (Lawless 2003), whereas the generalized log-logistic and Weibull PH models have relative risk interpretation. In the fourth example (Section 3.3.4), we consider joint models based on the generalized log-logistic, Weibull and piecewise-constant baseline hazard functions.

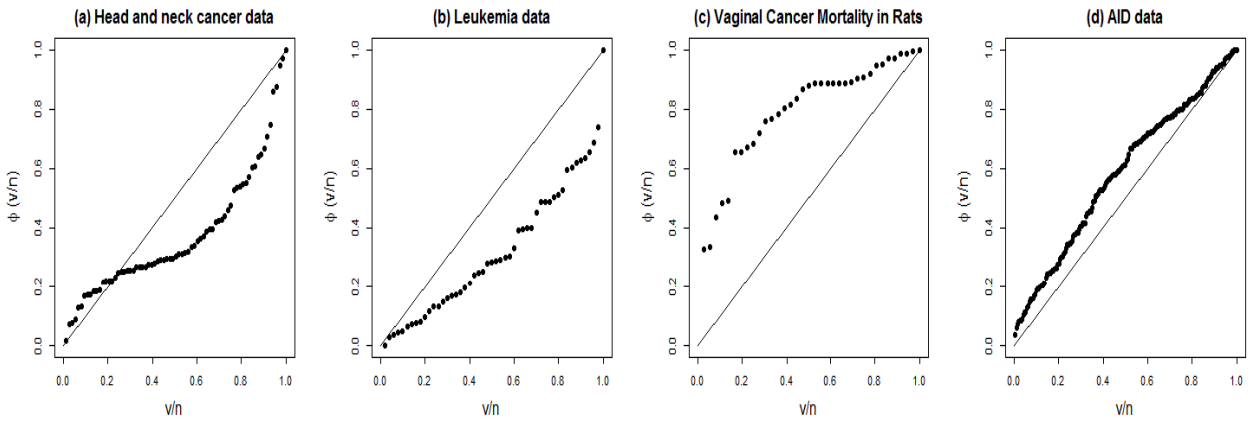

Figure 3.1: TTT plots for the four data sets used in Examples 1-4.

3.3.1 Example 1: Head and Neck Cancer Data

Data description, hazard shape and distributional assumption

Efron (1988) described a randomized clinical trial to compare radiation therapy alone (arm A) versus radiation plus chemotherapy (arm B) in treating head and neck cancer patients. Survival times (in days) for 51 patients in arm A (9 observations were censored) and 45 in arm B (14 were censored) were reported. The TTT plot in Figure 3.1(a) suggests a unimodal hazard shape of the survival times. Plots of $S(t;\hat{\boldsymbol{\theta}})$ and $\hat{S}(t)$ (Figure 3.2(a)-(c)) indicate more support for the generalized log-logistic distribution in comparison with the Weibull and log-logistic distributions in describing the head and neck cancer data.

Regression Analysis

Letting z_i = I(treatment = radiation therapy) that equals 1 if the treatment involves radiation therapy alone and 0 otherwise, we fit the generalized log-logistic PH, Weibull PH and log-logistic AFT models to the head and neck cancer data (numerical results are summarized in Table 3.1). The standard error of $\hat{\beta}$ for the generalized log-logistic model is smaller that those for the Weibull and log-logistic models, and therefore the generalized log-logistic would be preferred on grounds

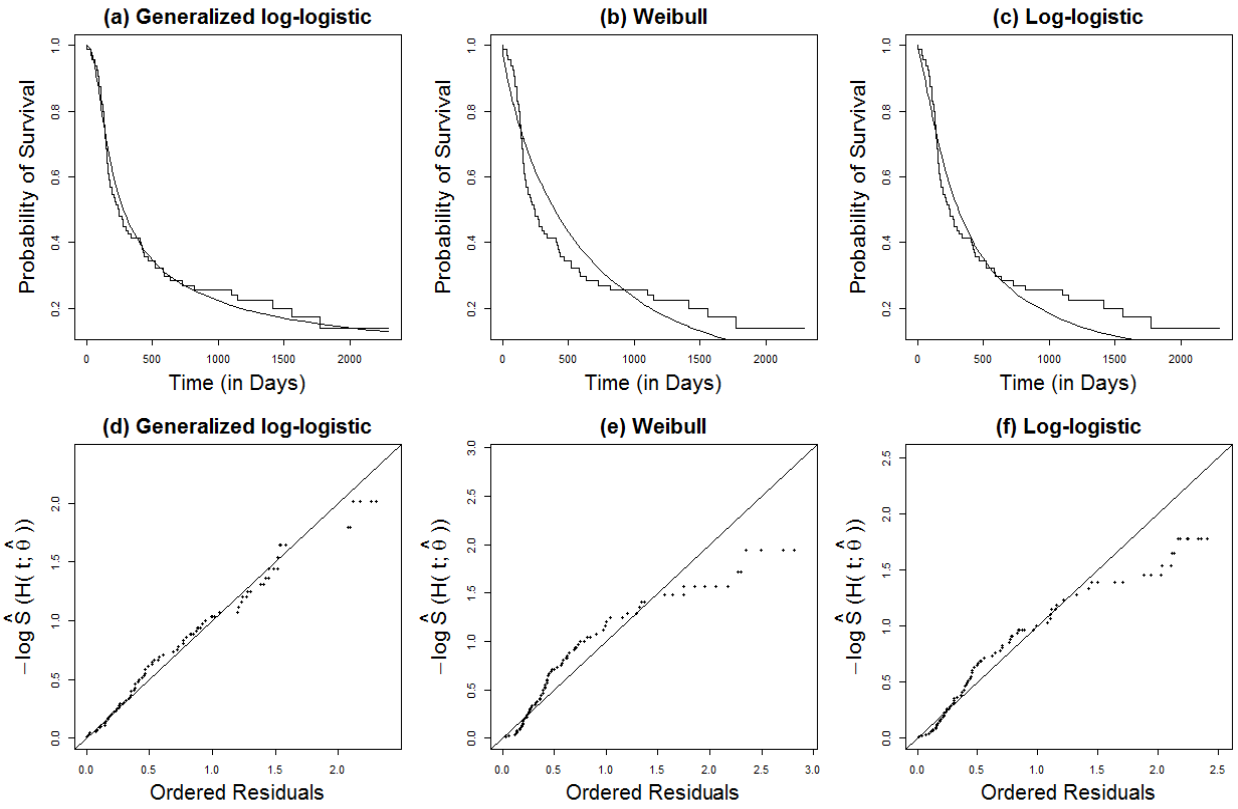

Figure 3.2: Diagnostic plots for the head and neck cancer data.

of efficiency. We also see that the generalized log-logistic has the lowest AIC value, which is supported by the residual plots (Figure 3.2(d)-(f)): residuals lying closely to the unit-slope line for generalized log-logistic indicate its superiority over the Weibull and log-logistic models. In summary, the generalized log-logistic fits the data adequately and is the best among the three models under consideration.

3.3.2 Example 2: Autologous and Allogeneic Bone Marrow Transplants

Data description, hazard shape and distributional assumption

Klein and Moeschberger (2003) described a study involving a sample of 101 patients with advanced acute myelogenous leukemia. Fifty-one of these patients had received an autologous (auto) bone marrow transplant, whereas 50 an allogeneic (allo) transplant. Survival times (in months) for

Table 3.1: Generalized log-logistic, Weibull and log-logistic fits for the head and neck cancer data.

	Generalized log-logistic PH (AIC = 1053.3934)		Weibull PH (AIC = 1082.5193)		Log-logistic AFT (AIC = 1067.2466)	
Parameter	Estimate	SE	Estimate	SE	Estimate	SE
β	0.5459	0.2382	0.6686	0.2415	−0.5549	0.2779
$\log \kappa$	0.9790	0.1986	−0.1619	0.0921	0.2764	0.0971
$\log \rho$	−5.2692	0.1844	−6.8248	0.2112	−6.0492	0.2128
$\log \gamma$	−4.6497	0.1755	–	–	–	–

28 auto transplant and 22 allo transplant patients were censored. Careful inspection of the TTT plot in Figure 3.1(b) reveals an indication of the unimodality of the hazard function. A comparison of the diagnostic plots (without covariate) in Figure 3.3(a)-(c) suggests that the assumption of generalized log-logistic is more appropriate than the assumption of Weibull or log-logistic in describing these data.

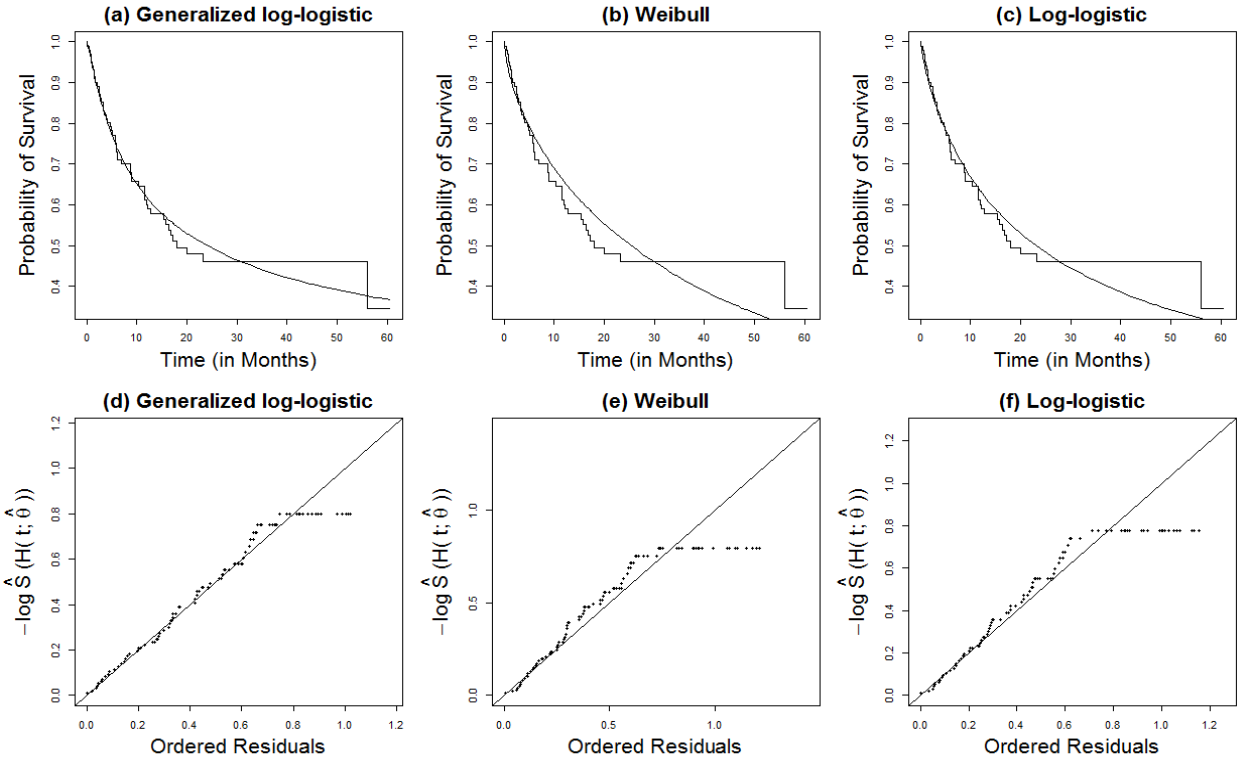

Figure 3.3: Diagnostic plots for the autologous and allogeneic bone marrow transplants data.

Regression Analysis

For regression analysis, we consider the covariate z_i = I(transplant = allo). The fits via the generalized log-logistic PH, Weibull PH and log-logistic AFT are summarized in Table 3.2. The generalized log-logistic has the lowest AIC value, suggesting it produced the best-fitting model. The residual plots (Figure 3.3(d)-(f)) also support this fact. It is interesting to note here that both the Weibull and log-logistic suggest a decreasing hazard function (estimate of the shape parameter is less than 1), whereas the generalized log-logistic captures the unimodal shape of the hazard function ($\hat{\kappa} = e^{0.2148} = 1.24 > 1$).

Table 3.2: Generalized log-logistic, Weibull and log-logistic fits for the bone marrow transplants data.

	Generalized log-logistic PH (AIC = 444.64)		Weibull PH (AIC = 450.08)		Log-logistic AFT (AIC = 446.46)	
Parameter	Estimate	SE	Estimate	SE	Estimate	SE
β	0.1981	0.2854	0.2535	0.2854	−0.0808	0.4481
$\log\kappa$	0.2148	0.2376	−0.3878	0.1229	−0.1694	0.1213
$\log\rho$	−2.4055	0.4917	−3.9683	0.3300	−3.1847	0.3474
$\log\gamma$	−1.3188	0.6253	–	–	–	–

3.3.3 Example 3: Vaginal Cancer Mortality in Rats

Data Description, hazard shape and distributional assumption

Pike (1966) described a laboratory experiment involving the development of vaginal cancer in rats insulted with the carcinogen DMBA. There were 19 rats in group 1 and 21 in group 2. Seventeen rats in group 1 and 19 in group 2 had developed tumours at the time the data were collected (i.e., two observations in each group were censored). There were reasonable scientific grounds for believing that there might be a threshold value before which no tumour could be detected. For this reason, Lawless (2003) considered the values $t' = t - 100$ to analyze these data. We also consider here the transformed version of the original observations. The TTT plot in Figure 3.1(c) suggests

an increasing hazard function for T'. Figure 3.4(a)-(c) shows diagnostic plots for the generalized log-logistic, Weibull and log-logistic fits (without covariate). We see that the generalized log-logistic and Weibull fits are similar, and provide slightly better description of the data compared to the log-logistic model.

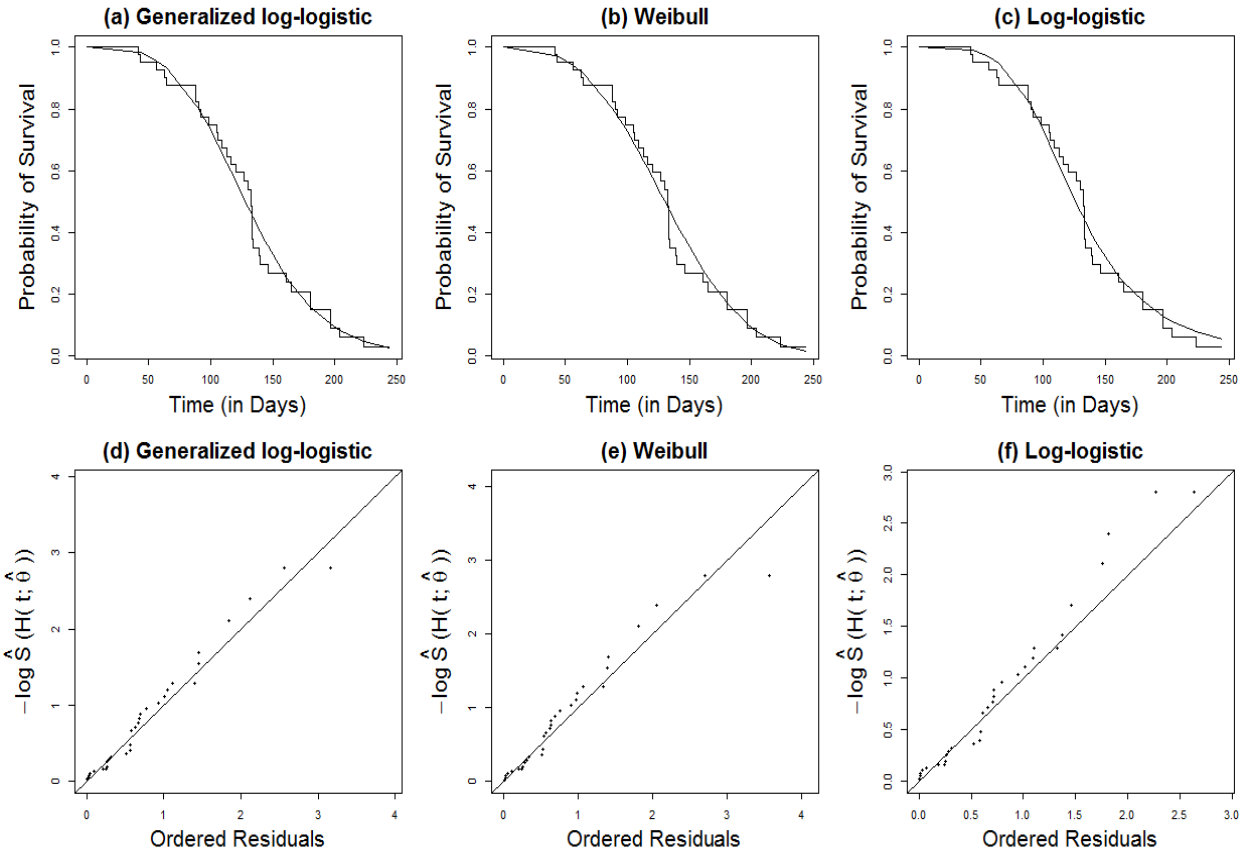

Figure 3.4: Diagnostic plots for the vaginal cancer mortality data.

Regression Analysis

For regression analysis, we consider the covariate z_i = I(group = group 1). Table 3.3 gives the estimates of the parameters and associated standard errors from the generalized log-logistic, Weibull and log-logistic fits. As demonstrated by the TTT plot, the Weibull PH suggests an increasing hazard rate ($\hat{\kappa} = e^{1.1308} = 3.098$). Note that a small value of $\hat{\gamma}$ ($\hat{\gamma} = e^{1.1308} = 0.007$) and

Table 3.3: Generalized log-logistic, Weibull and log-logistic fits for the vaginal cancer mortality data.

	Generalized log-logistic PH (AIC = 391.3523)		Weibull PH (AIC = 389.8671)		Log-logistic AFT (AIC = 391.8584)	
Parameter	Estimate	SE	Estimate	SE	Estimate	SE
β	0.6254	0.3485	0.6599	0.3474	−0.1861	0.1203
$\log\kappa$	1.2568	0.2168	1.1308	0.1300	1.5077	0.1429
$\log\rho$	−5.3516	0.5889	−5.0864	0.0754	−4.9301	0.0846
$\log\gamma$	−5.0190	0.1154	–	–	–	–

$\hat{\kappa} = e^{1.2568} = 3.514 > 1$ for generalized log-logistic also support this fact. Although the AIC values (Table 3.3) suggest no obvious preference of one model over the other, the residual plots (Figure 3.4(d)-(f)) clearly indicate more support for the generalized log-logistic and Weibull models. This example demonstrates that the generalized log-logistic has the ability to satisfactorily fit data which exhibit increasing hazard rates.

3.3.4 Example 4: AIDS Data

Data description and hazard Shape

This example illustrates the use of the generalized log-logistic distribution in joint modeling. Rizopoulos (2012) described a study involving 467 human immunodeficiency virus (HIV) infected patients who had failed or were intolerant to zidovudine therapy (ZT). The main objective was to compare two antiretroviral drugs to prevent the progression of HIV infections: didanosine (ddI) and zalcitabine (ddC). Patients were randomly assigned to receive either ddI or ddC and followed until death or the end of the study, resulted in 188 complete and 279 censored observations. It was also of interest to quantify the association between CD4 cell counts (internal time-dependent covariate) measured at $t = 0, 2, 6, 12$ and 18 months, and time to death. The TTT plot in Figure 3.1(d) indicates an increasing hazard shape.

Regression Analysis

For regression analysis, Rizopoulos (2012) considered joint models of the form

$$h_i(t; \mathbf{z}_i) = h_0(t; \boldsymbol{\alpha}) \exp\{\beta_0 + \beta_1 \text{drug}_i + \beta_2 \text{sex}_i + \beta_3 \text{ZT}_i + \beta_4 \text{CD4}_i(t)\}, \tag{3.11}$$

$$\text{CD4}_i(t) = \delta_0 + \delta_1 t + \delta_2 (t \times \text{drug}_i) + b_{0i} + b_{1i} t + \epsilon_i(t), \tag{3.12}$$

where (3.11) is the time-to-event model with $\text{drug}_i = \text{I}(\text{drug} = \text{ddI})$, $\text{sex}_i = \text{I}(\text{sex} = \text{male})$ and $\text{ZT}_i = \text{I}(\text{ZT} = \text{failure})$; and (3.12) is the longitudinal model with δ_0, δ_1 and δ_2 being the fixed-effects parameters, b_{0i} and b_{1i} the random-effects parameters, and $\epsilon_i(t)$ the random error component. We have reanalyzed the data here using generalized log-logistic, Weibull and piecewise-constant (six knots placed at equally spaced percentiles of the observed event times (Rizopoulos 2012)) baseline hazard functions in (3.11). Note that $h_0(t; \boldsymbol{\alpha}) = \kappa t^{\kappa-1}/[1 + (\gamma t)^{\kappa}]$ and $\kappa t^{\kappa-1}$ for generalized log-logistic and Weibull, respectively, and so $\beta_0 = \kappa \log \rho$ for both these models. For piecewise-constant baseline hazard, $h_0(t; \boldsymbol{\alpha}) = \sum_{q=1}^{7} \xi_q \text{I}(v_{q-1} < t \leq v_q)$ and $\beta_0 = 0$ in (3.11), where $0 = v_0 < v_1 < \ldots < v_7$ is the split of the time scale and ξ_q is the value of the hazard in the interval $(v_{q-1}, v_q]$. The estimates of the parameters and standard errors for the time-to-event process are presented in Table 3.4. We see that the estimates of the coefficients (i.e., $\hat{\beta}_1$, $\hat{\beta}_2$, $\hat{\beta}_3$ and $\hat{\beta}_4$) and their standard errors are broadly similar under the three competing models. The AIC values and residual plots (Figure 3.5) also suggest no obvious preference of one model over the other. Although we see no obvious preference of the generalized log-logistic model for this example for which the hazard function is monotone increasing, generalized log-logistic could be useful in joint modeling where the shape of the hazard function is unimodal.

Table 3.4: AIDS data: estimates and standard errors for the time-to-event process of joint models.

	Generalized log-logistic (AIC = 8699.611)		Weibull (AIC = 8699.258)		Piecewise-constant (AIC = 8711.614)	
Parameter	Estimate	SE	Estimate	SE	Estimate	SE
β_0	−3.1615	0.4411	−2.9477	0.3898	–	–
β_1	0.3690	0.1575	0.3727	0.1576	0.3647	0.1573
β_2	−0.3647	0.2583	−0.3619	0.2591	−0.3364	0.2585
β_3	0.3372	0.1556	0.3455	0.1555	0.3329	0.1555
β_4	−0.2824	0.0382	−0.2784	0.0378	−0.2860	0.0382
$\log \kappa$	0.3838	0.7709	0.2377	0.0732	–	–
$\log \gamma$	−2.8874	0.1333	–	–	–	–

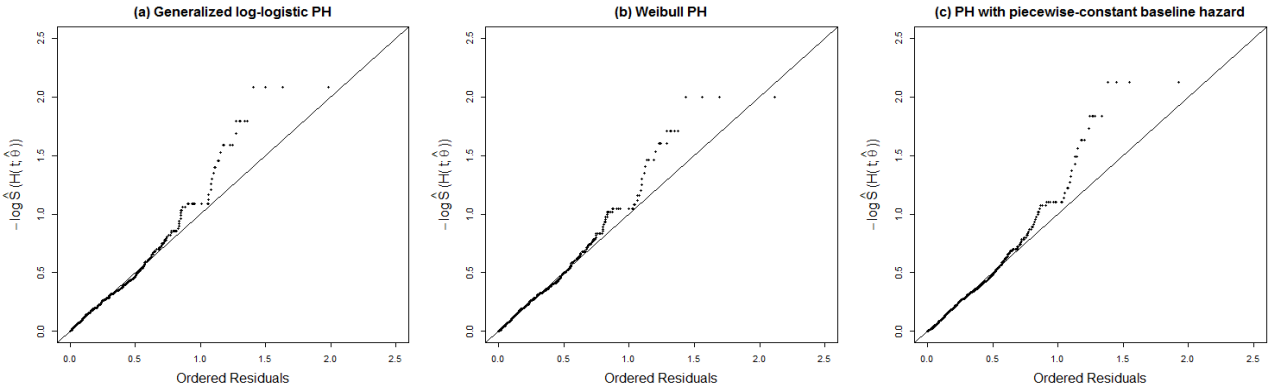

Figure 3.5: Residual plots for the AIDS data.

3.4 Simulation

Four covariates in a PH regression framework were considered in all simulations: two continuous covariates (z_1 and z_2), each generated from the standard normal distribution; and two binary covariates (z_3 and z_4), each generated from the *Bernoulli*(0.5) distribution. Regression parameter values were chosen to be $\boldsymbol{\beta} = (0.50, -0.50, 0.75, -0.75)'$ corresponding to the covariate vector $\mathbf{z} = (z_1, z_2, z_3, z_4)'$. To evaluate the performance of the generalized log-logistic model, we considered three simulation scenarios based on the shape of the hazard function. For each scenario (see below), lifetime data were generated from the generalized Weibull distribution with probability

density function

$$f(t;\boldsymbol{\alpha},\boldsymbol{\beta}) = \kappa\rho(\rho t)^{\kappa-1}\exp(\mathbf{z}'\boldsymbol{\beta})[1-\gamma(\rho t)^{\kappa}]^{\frac{\exp(\mathbf{z}'\boldsymbol{\beta})}{\gamma}-1}, \tag{3.13}$$

where $\rho > 0, \kappa > 0$ and $-\infty < \gamma < \infty$ are distributional parameters and $\boldsymbol{\alpha} = (\kappa, \gamma, \rho)'$; the support of the distribution is $t > 0$ for $\gamma \leq 0$ and $0 < t < \frac{1}{\rho\gamma^{\kappa}}$ for $\gamma > 0$. Generalized Weibull distribution includes all four basic shapes of hazard function: increasing for $\kappa \geq 1$ and $\gamma \geq 1$, decreasing for $0 < \kappa \leq 1$, unimodal for $\kappa > 1$ and $\gamma < 0$ and bathtub shape for $\kappa < 1$ and $\gamma > 0$. The simulation scenarios are then specified as follows.

1. **Scenario 1: Decreasing hazard.** Lifetimes were generated from the generalized Weibull with $\kappa = 0.5$, $\gamma = -0.1$ and $\rho = 0.1$, and censoring times were generated from the exponential distribution with rate parameter $\lambda = 0.045$.

2. **Scenario 2: Increasing Hazard.** Lifetimes were generated from the generalized Weibull with $\kappa = 2$, $\gamma = 0.1$ and $\rho = 0.1$, and censoring times were generated from the exponential distribution with rate parameter $\lambda = 0.060$.

3. **Scenario 3: Unimodal hazard.** Lifetimes were generated from the generalized Weibull with $\kappa = 2$, $\gamma = -0.1$ and $\rho = 0.1$ and censoring times were generated from the exponential distribution with rate parameter $\lambda = 0.060$.

Our choice of the parameter values led to, on average, 39.99, 40.69 and 42.99% censored observations for Scenarios 1-3, respectively. Given the covariates and censoring indicator, we then fit the generalized log-logistic, Weibull and Cox PH models to the simulated lifetimes. Note that since the Cox model is robust (usually fits the data well no matter which parametric model is appropriate), we consider this in our simulation study to compare model performance. For each scenario, 500 data sets (each of size n = 100) were generated, and the average of each of the estimated model parameters across these data sets was calculated. Absolute bias (AB) and mean square error (MSE) were then computed for model comparison (numerical results are summarized in Table 3.5).

Table 3.5: Model performance and comparison using simulation study ($n = 100$) with about 40% censored observations.

Scenarios	Parameter	True	Generalized log-logistic PH			Weibull PH			Cox PH		
			Mean	AB	MSE	Mean	AB	MSE	Mean	AB	MSE
Scenario 1	β_1	0.50	0.524	0.024	0.024	0.531	0.031	0.024	0.524	0.024	0.025
(True Model:	β_2	−0.50	−0.538	0.038	0.032	−0.545	0.045	0.033	−0.536	0.036	0.032
Generalized	β_3	0.75	0.798	0.048	0.088	0.808	0.058	0.090	0.795	0.045	0.091
Weibull)	β_4	−0.75	−0.780	0.030	0.081	−0.792	0.042	0.083	−0.782	0.032	0.083
	ρ	0.10	0.148	–	–	0.103	–	–	–	–	–
	κ	0.50	0.550	–	–	0.508	–	–	–	–	–
	γ	−0.10	0.073	–	–	–	–	–			
Scenario 2	β_1	0.50	0.516	0.016	0.027	0.518	0.018	0.027	0.532	0.032	0.031
(True Model:	β_2	−0.50	−0.522	0.022	0.035	−0.523	0.023	0.035	−0.533	0.033	0.039
Generalized	β_3	0.75	0.785	0.035	0.099	0.788	0.038	0.099	0.813	0.063	0.112
Weibull)	β_4	−0.75	−0.765	0.015	0.082	−0.768	0.018	0.082	−0.796	0.046	0.092
	ρ	0.10	0.107	–	–	0.106	–	–	–	–	–
	κ	2.00	2.269	–	–	2.249	–	–	–	–	–
	γ	0.10	0.006	–	–	–	–	–	–	–	–
Scenario 3	β_1	0.50	0.519	0.019	0.025	0.530	0.030	0.026	0.516	0.016	0.026
(True Model:	β_2	−0.50	−0.548	0.048	0.038	−0.557	0.057	0.039	−0.547	0.047	0.039
Generalized	β_3	0.75	0.791	0.041	0.099	0.811	0.061	0.103	0.791	0.041	0.102
Weibull)	β_4	−0.75	−0.790	0.040	0.089	−0.811	0.061	0.093	−0.792	0.042	0.095
	ρ	0.10	0.103	–	–	0.098	–	–	–	–	–
	κ	2.00	2.130	–	–	2.016	–	–	–	–	–
	γ	−0.10	0.024	–	–	–	–	–	–	–	–

Results for scenario 1

For the continuous covariates (z_1 and z_2), all three models produced estimates with similar MSE, whereas for the binary covariates (z_3 and z_4), the generalized log-logistic demonstrated the smallest MSE. In terms of bias, generalized log-logistic and Cox PH were roughly equivalent, and both were superior to Weibull.

Results for scenario 2

For the regression coefficients, the generalized log-logistic produced estimates with the smallest bias. We also see that the generalized log-logistic and Weibull produced estimates with similar

MSE, and both were superior to the Cox PH model. Note that the generalized log-logistic estimates for κ and γ were 2.269 and 0.006, respectively (i.e., the estimate of γ^{κ} is close to zero), supporting the fact that the hazard function is monotone increasing.

Results for scenario 3

In terms of bias, the generalized log-logistic and Cox PH produced comparable estimates of the regression coefficients. However, the generalized log-logistic produced the most accurate estimates in terms of MSE, mostly as a consequence of smaller standard deviations of the estimates. As expected, the Weibull produced the least accurate estimates in terms of both bias and MSE for Scenario 3 (i.e., unimodal hazard).

A simulation study with about 20% censored observations per data set also led to similar conclusions (Table 3.6). In summary, our simulation study has demonstrated that the generalized log-logistic could potentially be a very useful parametric model to adequately describe different types of time-to-event data.

Table 3.6: Model performance and comparison using simulation study ($n = 100$) with about 20% censored observations.

Scenarios	Para-meter	True	Generalized log-logistic PH			Weibull PH			Cox PH		
			Mean	AB	MSE	Mean	AB	MSE	Mean	AB	MSE
Scenario 1	β_1	0.50	0.519	0.019	0.023	0.524	0.024	0.023	0.522	0.022	0.024
(True Model:	β_2	−0.50	−0.535	0.035	0.030	−0.540	0.040	0.031	−0.535	0.035	0.031
Generalized	β_3	0.75	0.768	0.018	0.086	0.777	0.027	0.089	0.771	0.021	0.088
Weibull)	β_4	−0.75	−0.792	0.042	0.083	−0.802	0.052	0.085	−0.797	0.047	0.085
	ρ	0.10	0.151	–	–	0.116	0.016	0.004	–	–	–
	κ	0.50	0.552	–	–	0.518	0.018	0.003	–	–	–
	γ	−0.10	0.058	–	–	–	–	– –	–	–	
Scenario 2	β_1	0.50	0.515	0.015	0.022	0.520	0.020	0.022	0.520	0.020	0.023
(True Model:	β_2	−0.50	−0.524	0.024	0.032	−0.528	0.028	0.032	−0.526	0.026	0.032
Generalized	β_3	0.75	0.757	0.007	0.090	0.767	0.017	0.092	0.764	0.014	0.095
Weibull)	β_4	−0.75	−0.767	0.017	0.083	−0.777	0.027	0.084	−0.775	0.025	0.091
	ρ	0.10	0.104	–	–	0.102	0.002	0.000	–	–	–
	κ	2.00	2.165	–	–	2.112	0.112	0.059	–	–	–
	γ	0.10	0.013	–	–	–	–	–	–	–	–
Scenario 3	β_1	0.50	0.544	0.044	0.029	0.613	0.113	0.045	0.534	0.034	0.028
(True Model:	β_2	−0.50	−0.529	0.029	0.029	−0.574	0.074	0.037	−0.519	0.019	0.028
Generalized	β_3	0.75	0.778	0.028	0.100	0.891	0.141	0.148	0.760	0.010	0.098
Weibull)	β_4	−0.75	−0.790	0.040	0.088	−0.926	0.176	0.138	−0.774	0.024	0.084
	ρ	0.10	0.100	0.000	0.000	0.075	–	–	–	–	–
	κ	2.00	3.142	0.142	0.341	1.976	–	–	–	–	–
	γ	−0.10	0.094	0.006	0.001	–	–	–	–	–	–

3.5 Conclusion

In this Chapter, we proposed a simple extension of the log-logistic distribution to a PH model by appending an additional parameter. As described in Section 3.2, the proposed model naturally accommodates decreasing and unimodal hazard functions. The log-logistic distribution is known to be useful to describe unimodal hazard functions (Lawless 2003). As demonstrated in Examples 1 and 2, it turns out that the generalized log-logistic may provide better fits in describing unimodal hazard functions compared to the log-logistic distribution. Moreover, our simulation study revealed that the generalized log-logistic could produce more accurate results compared to the Weibull and

Cox PH models in describing monotone hazard functions. In summary, the flexibility provided by the generalized log-logistic model could be very useful in adequately describing different types of time-to-event data.

In the next chapter (Chapter 4), we present our second proposed distribution which accommodates both monotone and nonmonotone hazards and based on the proposed distribution we develop the flexible parametric PH model. Then the proposed model has been tailored towards the recurrent events data analysis and joint modelling of longitudinal and time-to event data in Chapter 5.

Chapter 4

THE MODIFIED KUMARASWAMY WEIBULL DISTRIBUTION: A FLEXIBLE PROPORTIONAL HAZARDS MODEL FOR TIME-TO-EVENT DATA

In chapter 3, we proposed a generalization of the log-logistic distribution, which is closed under the proportional hazard assumption. Then, a proportional hazard regression model for time-to-event data was formulated based on the proposed distribution. The generalized log-logistic proportional hazard model naturally accommodates monotone decreasing and unimodal hazard functions, and has the ability to model increasing hazard shapes satisfactorily (the model approaches the Weibull in the limit). Aside from the fact that it does not naturally accommodate monotone increasing hazard shapes, the generalized log-logistic model is also not flexible enough to deal with bathtub-shaped hazard functions. The bathtub curve assumes a continuous function comprised of three segments connected together: a decreasing failure rate in the incoming phase and an increasing failure rate in the outgoing phase, joined by an approximately constant and lower rate to model a normal life period (commonly known as "useful life"). This type of shape is widely used to describe data in reliability engineering. For example, The failure rate in the early life of an electronic unit or system is usually high perhaps because of faulty design and/or defective items, but it rapidly decreases as defective items are identified and discarded or reworked, leading to a low and (approximately) constant failure rate in the mid-life of the unit. However, the failure rate increases after the mid-life due to wear-out of the system. Such a process can be described by a bathtub-

shaped hazard function. Note that this general concept is also applicable to describe the process of human life: the risk is high (high infant mortality) during an initial period, after which the hazard function stays approximately constant and low until a certain time, and then it increases because of aging. In this Chapter, we propose a more general parametric proportional hazards model, which is parsimonious and flexible in the sense that it accommodates all four standard shapes of the hazard function (increasing, decreasing, unimodal and bathtub shape) at the small cost of estimating only three distributional parameters. We then formulate a regression model based on the proposed distribution, and develop large sample theory for statistical inference. A simulation study and real data example reveal that the proposed model can be valuable in adequately describing different types of time-to-event data. Applications of the proposed distribution in recurrent event data analysis and joint modeling of time-to-event and longitudinal data are considered in Chapter 5. An article based on the findings of this study (Chapters 4 and 5) is in progress.

In Section 4.1, we present the context and motivation of this study. The proposed model is derived from the Kumaraswamy Weibull (KumW) distribution of Cordeiro et al. (2010), which is described in Section 4.2. In Section 4.3, we propose the modified Kumaraswamy Weibull (MKumW) distribution and present some of its properties, including the fact that it is closed under the proportional hazards family. For regression analysis, we then formulate the proportional hazards model based on the MKumW distribution (Section 4.4). The maximum likelihood estimation, large sample theory for inference and methods to check model assumptions are presented in Section 4.5. In Section 4.8, we demonstrate the performance of the proposed model with applications to both real and simulated data. We conclude in Section 4.10 by summarizing our findings.

4.1 Introduction

Proportional hazards (PH) models are widely used in survival and recurrent event data analyses (Cook and Lawless 2007, Kalbfleisch and Prentice 2002, Kleinbaum and Klein 2012) as well

as in joint modelling of longitudinal and time-to-event-data (Rizopoulos 2012). A PH model is commonly expressed using the hazard function of a non-negative random variable T (time to an event) as follows:

$$h(t; \mathbf{z}) = h_0(t)\psi(\mathbf{z}), \tag{4.1}$$

where $\mathbf{z}$ is a column vector of covariates, and $h_0(t)$ and $\psi(\mathbf{z})$ are positive-valued functions, commonly called the baseline hazard function (i.e., hazard function for a subject with covariate vector $\mathbf{z} = \mathbf{0}$) and regression function, respectively.

A PH model can be formulated with or without assuming a parametric model for $h_0(t)$: the semi-parametric Cox PH model assumes that $h_0(t)$ is an arbitrary positive-valued function of t (Cox 1972), whereas a fully parametric model is formulated by using a parametric form for $h_0(t)$ (Lawless 2003). The Cox PH is perhaps the most widely used regression model in survival analysis because of (1) the semi-parametric nature of the model (i.e., no distributional assumption is required for T), (2) the robustness property (the Cox model usually fits the data well no matter which parametric model is appropriate (Kleinbaum and Klein 2012)), and (3) the availability of statistical software packages to fit the model. However, the Cox model has some intrinsic features that may lead to inefficient statistical inference and/or loss of information, as outlined below.

- Parametric PH models may lead to more efficient estimates than the semi-parametric Cox model under certain conditions (Efron 1988, Oakes 1977, Royston and Parmar 2002).

- An adequate parametric model may lead to more precise estimates of the survival probabilities and related quantities (Hjort 1992).

- The baseline hazard function of the Cox model is regarded as a nuisance parameter, which is of fundamental interest in medical research due to its direct association with the time-course of illness.

- The use of Cox PH in joint modeling of time-to-event and longitudinal data (Wulfsohn and

Tsiatis 1997) usually leads to an underestimation of the standard errors of the parameter estimates (Hsieh et al. 2006, Rizopoulos 2012), and therefore most methods for joint modeling are based on parametric response distributions (Hwang and Pennell 2014).

Since a fully specified model is often more reliable for analyzing complex data structure and processes, Cox also expressed his preference towards using parametric models (Reid 1994). Note that in addition to calculating relative effect estimates, a parametric model can also be used to predict survival time, hazard rates, and mean and median survival times.

There are only a few distributions that are closed under the PH assumption (e.g., Weibull and generalized log-logistic distributions). There is also a trade-off between parsimony and flexibility among the parametric PH models. For example, the Weibull PH model is parsimonious (involves only two distributional parameters) but not flexible in the sense that it accommodates only monotone increasing and decreasing hazard functions (Lawless 2003). The generalized log-logistic distribution has three parameters but accommodates only monotone decreasing and unimodal hazard shapes, and can handle increasing hazard function satisfactorily (Khan and Khosa 2016). The lognormal-power function distribution with four parameters accommodates monotone decreasing, unimodal and bathtub shape hazard rates (Reed 2011). A more complex approach of using natural cubic splines to smooth the baseline hazard function under the Weibull regression setup is also considered (Royston and Parmar 2002). There is only one three-parameter PH model, viz. the generalized Weibull distribution (Mudholkar et al. 1996), that accommodates all four standard shapes of the hazard function (increasing, decreasing, unimodal and bathtub shape). However, one difficulty with this model is that it is nonregular (the support depends on some parameters) for increasing and bathtub-shaped hazard functions, and therefore the standard maximum likelihood asymptotics do not hold. The lack of parsimonious and flexible parametric models to characterize different types of time-to-event data could be another reason for the extensive use of the Cox PH model in survival analysis.

We propose a three-parameter model based on a modification of the Kumaraswamy Weibull

(KumW) distribution of Cordeiro et al. (2010). The proposed model is closed under the proportionality of hazards, and is flexible and parsimonious in the sense that it accomodates all four basic shapes of the hazard function at the small cost of estimating three distributional parameters. The KumW distribution is described in Section 4.2, and the proposed model is presented in Section 4.3.

4.2 The Kumaraswamy Weibull Distribution

For a positive-valued random variable T, the probability density function of the Kumaraswamy-G Distribution (Cordeiro and de Castro 2011) is given by

$$f(t) = \gamma\rho g(t)[G(t)]^{\gamma-1}\{1-[G(t)]^{\gamma}\}^{\rho-1}, \tag{4.2}$$

where $G(t)$ in (4.2) is an arbitrary cumulative distribution function, $g(t) = dG(t)/dt$, and $\gamma > 0$ and $\rho > 0$ are the parameters. The Kumaraswamy Weibull (KumW) distribution (Cordeiro et al. 2010) is defined using the Weibull cumulative distribution function $G(t) = 1 - \exp(-\lambda t^{\kappa})$ so that

$$f(t) = \kappa\gamma\rho\lambda\{1-[1-\exp(-\lambda t^{\kappa})]^{\gamma}\}^{\rho-1}\left[1-\exp(-\lambda t^{\kappa})\right]^{\gamma-1}\exp(-\lambda t^{\kappa})t^{\kappa-1}, \tag{4.3}$$

$$S(t) = \{1-[1-\exp(-\lambda t^{\kappa})]^{\gamma}\}^{\rho}, \tag{4.4}$$

$$h(t) = \frac{\kappa\gamma\rho\lambda[1-\exp(-\lambda t^{\kappa})]^{\gamma-1}\exp(-\lambda t^{\kappa})t^{\kappa-1}}{1-[1-\exp(-\lambda t^{\kappa})]^{\gamma}}, \tag{4.5}$$

Note that when $\rho = 1$, (4.5) reduces to

$$h(t) = \frac{\kappa\gamma\lambda[1-\exp\{-\lambda t^{\kappa}\}]^{\gamma-1}\exp(-\lambda t^{\kappa})t^{\kappa-1}}{1-[1-\exp\{-\lambda t^{\kappa}\}]^{\gamma}}, \tag{4.6}$$

which is the hazard function of the exponentiated Weibull distribution as described in Section 2.4.7. The Weibull and the exponentiated exponential distributions are also sub-models of the KumW distribution with $\rho = \gamma = 1$ and $\rho = \kappa = 1$, respectively.

4.3 The Modified Kumaraswamy Weibull Distribution

We propose the Modified Kumaraswamy Weibull (MKumW) distribution as special case of the Kumaraswamy Weibull distribution by taking $\lambda = 1$ in (4.3), so that

$$f(t) = \kappa\gamma\rho \{1 - [1 - \exp(-t^{\kappa})]^{\gamma}\}^{\rho-1} [1 - \exp(-t^{\kappa})]^{\gamma-1} \exp(-t^{\kappa}) t^{\kappa-1}, \tag{4.7}$$

$$S(t) = \{1 - [1 - \exp(-t^{\kappa})]^{\gamma}\}^{\rho}, \tag{4.8}$$

$$h(t) = \frac{\kappa\gamma\rho [1 - \exp(-t^{\kappa})]^{\gamma-1} \exp(-t^{\kappa}) t^{\kappa-1}}{1 - [1 - \exp(-t^{\kappa})]^{\gamma}}, \tag{4.9}$$

$$H(t) = -\rho \log\{1 - [1 - \exp(-t^{\kappa})]^{\gamma}\}, \tag{4.10}$$

where $\kappa > 0$ and $\gamma > 0$ are the shape parameters and $\rho > 0$ is the inverse scale parameter. The distribution reduces to the Weibull family when $\gamma = 1$. The proposed model is parsimonious compared to the KumW distribution; it is also flexible enough to accommodate all four basic shapes of the hazard function as described below (Theorem 1).

The median of the MKumW distribution is

$$M = \{-\log[1 - (1 - (\frac{1}{2})^{1/\rho})^{1/\gamma}]\}^{1/\kappa}, \tag{4.11}$$

and the r^{th} moment is

$$E(T^r) = \rho \int_0^1 \{-\log[1 - (1 - u)^{1/\gamma}]\}^{r/\kappa} u^{\rho-1} du, \tag{4.12}$$

where the integral in (4.12), in general, does not have an explicit solution. Figure 4.1 shows the density curves for T (along with their means and medians) with different choices of the parameter values. Note that numerical approximation is used to obtain the mean of the distribution as shown in Figure 4.1, and the R (R Core Team 2016) function *integrate* is used for this purpose.

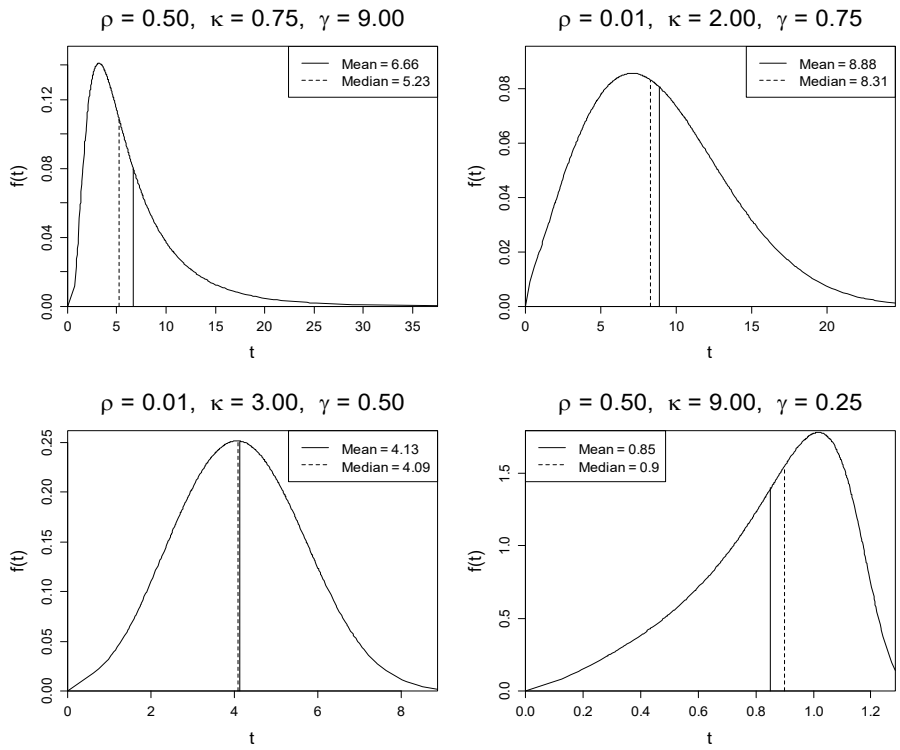

Figure 4.1: MKumW probability density functions for different values of the parameters κ, γ and ρ, where κ and γ are the shape parameters and ρ is the inverse scale parameter.

The MKumW hazard function (4.9) assumes both monotone and nonmonotone shapes as summarized by the following theorem.

Theorem 1. For the MKumW family (4.7), the hazard function is (a) monotone increasing for $\kappa \geq 1$ and $\kappa\gamma \geq 1$, (b) monotone decreasing for $\kappa \leq 1$ and $\kappa\gamma \leq 1$, (c) unimodal for $\kappa < 1$ and $\kappa\gamma > 1$, and (d) bathtub-shaped for $\kappa > 1$ and $\kappa\gamma < 1$.

To prove the theorem, recall that ρ and λ are the inverse scale parameters of the MKumW and the exponentiated Weibull distributions, respectively. Thus, these parameters have no effect on the shape of the hazard functions of these two distributions. Since the exponentiated Weibull and the MKumW hazard functions are identical when $\rho = 1$ and $\lambda = 1$ (see Equations (4.5) and (4.9)), the proof of the theorem for $\rho = 1$ and $\lambda = 1$ follows from *Theorem* 2.1 of Mudholkar et al. (1996). Since the theorem is true for $\rho = 1$, it is also true for all $\rho > 0$.

Typical shapes of the MKumW hazard functions for different combinations of the parameters

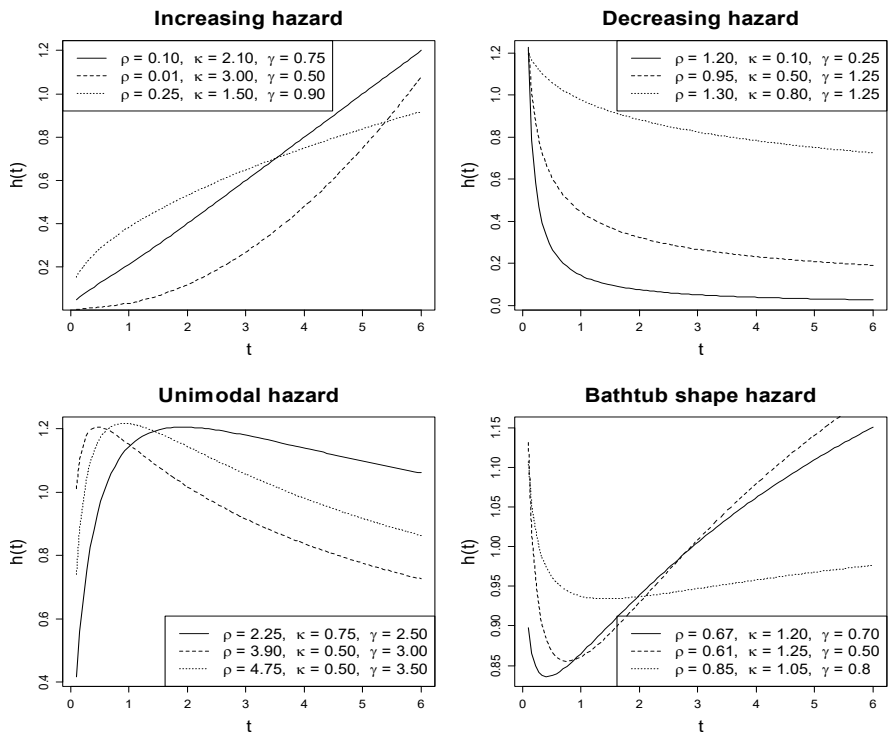

Figure 4.2: MKumW hazard functions for different values of the parameters κ, γ and ρ, illustrating the fact that the model accommodates all four basic shapes of the hazard function (increasing, decreasing, unimodal and bathtub shape).

are shown in Figure 4.2.

4.4 The MKumW Proportional Hazards Model

Using $\psi(\mathbf{z}) = \exp(\mathbf{z}'\boldsymbol{\beta})$ in (4.1), the MKumW regression model can be expressed as

$$h(t;\mathbf{z}) = h_0(t)\exp(\mathbf{z}'\boldsymbol{\beta}), \tag{4.13}$$

where $\boldsymbol{\beta} = (\beta_1,\beta_2,\ldots,\beta_p)'$ is the vector of regression coefficients associated with the covariate vector $\mathbf{z}$, and, $h_0(t)$ is the baseline hazard function given by (4.9). This formulation satisfies the requirement that $\psi(\mathbf{z}) > 0$ for all possible values of $\mathbf{z}$, and this is the form that is adopted in what follows. We see that the MKumW family is closed under the PH assumption, as (4.13) is

a member of the family (4.9) with $\rho^* = \rho \exp(\mathbf{z}'\boldsymbol{\beta})$. An appealing feature of (4.13) is that the regression coefficients have relative risk interpretation: a unit increase in z_j is associated with a multiplicative increase e^{β_j} in the hazard rate, leading to a change in relative risk as the covariate varies (Congdon 2014). We also see that the hazard ratio comparing any two specifications of covariates is independent of time, a feature which is known as the PH assumption (2.6.2).

4.5 Maximum Likelihood Method for Right Censored Data

Let there be n individuals with lifetimes denoted by $T_1, T_2, \ldots, T_n$. Assuming that the data are subject to right censoring, we observe $t_i = \min(T_i, C_i)$, where $C_i > 0$ corresponds to a potential censoring time for individual i. Letting $\delta_i = \mathrm{I}(T_i \leq C_i)$ that equals 1 if $T_i \leq C_i$ and 0 otherwise, the observed data for individual i consist of $\{t_i, \delta_i, \mathbf{z}_i\}$, $i = 1, 2, \ldots, n$, where t_i is a lifetime or censoring time according to whether $\delta_i = 1$ or 0, respectively, and $\mathbf{z}_i = (z_{i1}, z_{i2}, \ldots, z_{ip})'$ is a $p \times 1$ column vector of external covariates. Using (2.27), the log-likelihood contribution from individual i can be written as

$$\ell_i(\boldsymbol{\theta}) = \delta_i\{\log h_0(t_i) + \mathbf{z}_i'\boldsymbol{\beta}\} - \int_0^{t_i} h_0(u) \exp(\mathbf{z}_i'\boldsymbol{\beta}) du, \tag{4.14}$$

where $\boldsymbol{\theta}$ denotes all the model parameters collectively. For our model, $\boldsymbol{\theta}$ includes the regression coefficients $\boldsymbol{\beta}$ and the distributional parameters $\boldsymbol{\zeta} = (\kappa, \gamma, \rho)'$ that characterize the baseline hazard function $h_0(t)$, leading to $\boldsymbol{\theta} = (\boldsymbol{\zeta}', \boldsymbol{\beta}')'$. Using (4.9) for $h_0(\cdot)$ and noting that $H_0(t) = \int_0^t h_0(u)du$ is the baseline cumulative hazard function as given by (4.10), the full log-likelihood function can be expressed as

$$\begin{aligned}
\ell(\boldsymbol{\theta}) = {} & m \log \kappa + m \log \gamma + m \log \rho + (\gamma - 1) \sum_{i=1}^{n} \delta_i \log(1 - a_i) \\
& - \sum_{i=1}^{n} \delta_i \log[1 - (1 - a_i)^{\gamma}] + (\kappa - 1) \sum_{i=1}^{n} \delta_i \log t_i - \sum_{i=1}^{n} \delta_i t_i^{\kappa} + \sum_{i=1}^{n} \delta_i \mathbf{z}_i'\boldsymbol{\beta} \\
& + \rho \sum_{i=1}^{n} \log[1 - (1 - a_i)^{\gamma}] \exp(\mathbf{z}_i'\boldsymbol{\beta}),
\end{aligned} \tag{4.15}$$

where $m = \sum_{i=1}^{n} \delta_i$ and $a_i = \exp(-t_i^{\kappa})$. The maximum likelihood score equations are obtained by differentiating $\ell(\boldsymbol{\theta})$ with respect to the parameters, and are given by

$$\begin{aligned} \mathrm{U}_{\kappa}(\boldsymbol{\theta}) = \frac{\partial \ell(\boldsymbol{\theta})}{\partial \kappa} &= \frac{m}{\kappa} + \sum_{i=1}^{n} \delta_i \log t_i - \sum_{i=1}^{n} \delta_i t_i^{\kappa} \log t_i \\ &\quad + (\gamma - 1) \sum_{i=1}^{n} \delta_i \Big(\frac{a_i t_i^{\kappa} \log t_i}{1 - a_i} \Big) + \gamma \sum_{i=1}^{n} \delta_i \Big[\frac{a_i (1 - a_i)^{\gamma - 1}\, t_i^{\kappa} \log t_i}{1 - (1 - a_i)^{\gamma}} \Big] \\ &\quad - \gamma \rho \sum_{i=1}^{n} \Big[\frac{a_i (1 - a_i)^{\gamma - 1}\, t_i^{\kappa} \log t_i}{1 - (1 - a_i)^{\gamma}} \Big] \exp(\mathbf{z}_i' \boldsymbol{\beta}) = 0, \end{aligned} \tag{4.16}$$

$$\begin{aligned} \mathrm{U}_{\gamma}(\boldsymbol{\theta}) = \frac{\partial \ell(\boldsymbol{\theta})}{\partial \gamma} &= \frac{m}{\gamma} + \sum_{i=1}^{n} \delta_i \log(1 - a_i) + \sum_{i=1}^{n} \delta_i \Big[\frac{(1 - a_i)^{\gamma} \log(1 - a_i)}{1 - (1 - a_i)^{\gamma}} \Big] \\ &\quad - \rho \sum_{i=1}^{n} \Big[\frac{(1 - a_i)^{\gamma} \log(1 - a_i)}{1 - (1 - a_i)^{\gamma}} \Big] \exp(\mathbf{z}_i' \boldsymbol{\beta}) = 0, \end{aligned} \tag{4.17}$$

$$\mathrm{U}_{\rho}(\boldsymbol{\theta}) = \frac{\partial \ell(\boldsymbol{\theta})}{\partial \rho} = \frac{m}{\rho} + \sum_{i=1}^{n} \log[1 - (1 - a_i)^{\gamma}] \exp(\mathbf{z}_i' \boldsymbol{\beta}) = 0, \tag{4.18}$$

$$\mathrm{U}_{\beta_j}(\boldsymbol{\theta}) = \frac{\partial \ell(\boldsymbol{\theta})}{\partial \beta_j} = \sum_{i=1}^{n} \delta_i z_{ij} + \rho \sum_{i=1}^{n} z_{ij} \log[1 - (1 - a_i)^{\gamma}] \exp(\mathbf{z}_i' \boldsymbol{\beta}) = 0, \tag{4.19}$$

where $j = 1, 2, \ldots, p$. As indicated in Chapter 3, an unrestricted parameter space is usually preferred to improve the convergence of the iterative procedure for maximum likelihood estimation and the accuracy of large sample methods. We propose the parameterizations $\tau = -\log \kappa$, $\xi = \log \gamma$ and $\upsilon = \log \rho$, and denote the parameter vector by $\boldsymbol{\theta} = (\tau, \xi, \upsilon, \boldsymbol{\beta}')'$. Large sample theory for statistical inference is based on the $(3 + p) \times (3 + p)$ observed information matrix,

$$I(\boldsymbol{\theta}) = -\begin{bmatrix} \frac{\partial^2 \ell(\boldsymbol{\theta})}{\partial \tau^2} & \frac{\partial^2 \ell(\boldsymbol{\theta})}{\partial \tau \partial \xi} & \frac{\partial^2 \ell(\boldsymbol{\theta})}{\partial \tau \partial \upsilon} & \frac{\partial^2 \ell(\boldsymbol{\theta})}{\partial \tau \partial \beta_1} & \cdots & \frac{\partial^2 \ell(\boldsymbol{\theta}^*)}{\partial \tau \partial \beta_p} \\ \frac{\partial^2 \ell(\boldsymbol{\theta})}{\partial \xi \partial \tau} & \frac{\partial^2 \ell(\boldsymbol{\theta})}{\partial \xi^2} & \frac{\partial^2 \ell(\boldsymbol{\theta})}{\partial \xi \partial \upsilon} & \frac{\partial^2 \ell(\boldsymbol{\theta})}{\partial \xi \partial \beta_1} & \cdots & \frac{\partial^2 \ell(\boldsymbol{\theta})}{\partial \xi \partial \beta_p} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \frac{\partial^2 \ell(\boldsymbol{\theta})}{\partial \beta_p \partial \tau} & \frac{\partial^2 \ell(\boldsymbol{\theta})}{\partial \beta_p \partial \xi} & \frac{\partial^2 \ell(\boldsymbol{\theta})}{\partial \beta_p \partial \upsilon} & \frac{\partial^2 \ell(\boldsymbol{\theta})}{\partial \beta_p \partial \beta_1} & \cdots & \frac{\partial^2 \ell(\boldsymbol{\theta})}{\partial \beta_p^2} \end{bmatrix}.$$

Note that second derivatives of $\ell(\boldsymbol{\theta})$ are given in Appendix A.3. Variance estimates for the maximum likelihood estimators can be obtained from $I^{-1}(\hat{\boldsymbol{\theta}})$, where $\hat{\boldsymbol{\theta}}$ is the maximum likelihood estimate of $\boldsymbol{\theta}$. Since $(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}) \sim \mathrm{N}(\mathbf{0}, I^{-1}(\hat{\boldsymbol{\theta}}))$ for large samples, statistical inference is based on the asymptotic normality of the maximum likelihood estimator $\hat{\boldsymbol{\theta}}$. Note that the observed information matrix at $\hat{\boldsymbol{\theta}}$ can be evaluated using the *hessain* function of the numDeriv package (Gilbert and Varadhan 2016) in (R Core Team 2016).

Inference on the MKumW shape parameters κ and $\kappa\gamma$ are of particular interest, as these parameters determine the shape of the hazard function. The maximum likelihood estimate and confidence interval for κ can be obtained using $\hat{\kappa} = e^{-\hat{\tau}}$ and (e^{-U}, e^{-L}), respectively, where (L, U) is the confidence interval for τ. A point estimate of $\kappa\gamma$ is $\widehat{\kappa\gamma} = e^{-\hat{\tau}+\hat{\xi}}$. To find a confidence interval estimate for $\kappa\gamma$, we first compute the confidence interval for $\log(\kappa\gamma) = -\tau + \xi$ using $\widehat{\log(\kappa\gamma)} = -\hat{\tau} + \hat{\xi}$ and $\mathrm{SE}(-\hat{\tau} + \hat{\xi}) = [\widehat{\mathrm{var}}(\hat{\tau}) + \widehat{\mathrm{var}}(\hat{\xi}) - 2\widehat{\mathrm{cov}}(\hat{\tau}, \hat{\xi})]^{1/2}$. A confidence interval for $\kappa\gamma$ can then be calculated by exponentiating the confidence interval for $\log(\kappa\gamma)$.

4.5.1 Computation and Initial Values

An optimization software such as the R (R Core Team 2016) function *optim* or *nlminb* can be used to find $\hat{\boldsymbol{\theta}}$ that minimizes the negative log-likelihood function (i.e., maximizes the log-likelihood function). Although the specification of the derivatives is optional in these R functions, fast and rapid convergence may be achieved if the expressions for the negative log-likelihood function are

provided. The observed information matrix can be obtained either directly evaluating the second derivatives at $\hat{\boldsymbol{\theta}}$ or by numerical differentiation. Note that the function *optim* can produce the information matrix $-I(\hat{\boldsymbol{\theta}})$ using numerical differentiation if requested, whereas the function *hessian* in R package "numDeriv" (Gilbert and Varadhan 2016) may be used to obtain this matrix if *nlminb* is used. In our implementation (R codes are given in Appendix A.4), the observed information matrix is obtained using the *hessian* function of the numDeriv package in R. An optimization algorithm requires a set of initial values for the parameters. We see from (4.18) that

$$\rho = \frac{m}{-\sum_{i=1}^{n} \log[1-(1-a_i)^{\gamma}] \exp(\mathbf{z}_i'\boldsymbol{\beta})}. \tag{4.20}$$

Substituting this in (4.15), the profile log-likelihood function can be written as

$$\begin{aligned}\ell_p(\kappa,\gamma,\boldsymbol{\beta}) \propto\ & m\log\kappa + m\log\gamma - m\log\Big\{-\sum_{i=1}^{n}\log[1-(1-a_i)^{\gamma}]\exp(\mathbf{z}_i'\boldsymbol{\beta})\Big\} \\ & + (\gamma-1)\sum_{i=1}^{n}\delta_i\log(1-a_i) - \sum_{i=1}^{n}\delta_i\log[1-(1-a_i)^{\gamma}] \\ & + (\kappa-1)\sum_{i=1}^{n}\delta_i\log t_i - \sum_{i=1}^{n}\delta_i t_i^{\kappa} + \sum_{i=1}^{n}\delta_i\mathbf{z}_i'\boldsymbol{\beta}. \end{aligned} \tag{4.21}$$

For the regression coefficients $\boldsymbol{\beta}$, we propose to use the Cox PH estimate $\tilde{\boldsymbol{\beta}}$ as initial values because of the robustness property of the Cox model. Using $\tilde{\boldsymbol{\beta}}$ for $\boldsymbol{\beta}$, we then obtain $\tilde{\tau}$ and $\tilde{\eta}$ that maximizes (4.21). Once $\tilde{\tau}$, $\tilde{\xi}$ and $\tilde{\boldsymbol{\beta}}$ are obtained, we find $\tilde{\upsilon} = \widetilde{\log\rho}$ using (4.20). Finally, $(\tilde{\tau},\tilde{\xi},\tilde{\upsilon},\tilde{\boldsymbol{\beta}})'$ are used as initial values to maximize $\ell(\boldsymbol{\theta})$. Note that a few terms in the log-likelihood function are numerically difficult to compute, as R uses only double-precision floating point numbers (floating-point numbers are more dense near zero). For example, if $\exp(-t^{\kappa})$ is very close to zero, then $\log\{1-[1-\exp(-t^{\kappa})]^{\gamma}\}$ may lead to undefined results in R. To deal with this problem, we used the R package *Rmpfr* (Maechler 2016) to obtain high precision numbers.

4.5.2 Goodness of Fit of the Weibull as a Submodel

Recall that $\gamma = 1$ (or equivalently $\xi = 0$) reduces the MKumW to Weibull model. To test the goodness of fit of Weibull as a submodel, we can use the likelihood ratio test, defined by

$$\Lambda = 2\ell(\hat{\tau}, \hat{\xi}, \hat{\upsilon}, \hat{\boldsymbol{\beta}}) - 2\ell(\tilde{\tau}, \eta = 0, \tilde{\upsilon}, \tilde{\boldsymbol{\beta}}),$$

where $\hat{\tau}$, $\hat{\xi}$, $\hat{\upsilon}$ and $\hat{\boldsymbol{\beta}}$ are the maximum likelihood estimates under the MKumW model, and $\tilde{\tau}$, $\tilde{\upsilon}$ and $\tilde{\boldsymbol{\beta}}$ are the maximum likelihood estimates under H_0: $\xi = 0$ (i.e., Weibull PH). For large samples, υ is approximately distributed as $\chi^2_{(1)}$ under H_0.

4.6 Model Diagnostics

Testing of a parametric PH model's goodness of fit consists of (1) inspection of residuals to check the appropriateness of the underlying probability distribution, and (2) use of statistical tests and/or graphical procedures to check the PH assumption. These two components of the goodness of fit in the context of the MKumW model are described in this section.

4.6.1 Residual Plot to Check the Assumption of MKumW

Hazard based residuals are defined by $H(t_i|\mathbf{z}_i, \hat{\boldsymbol{\theta}}) = H_0(t_i|\hat{\kappa}, \hat{\gamma}, \hat{\rho})$, where $H_0(t_i|\hat{\kappa}, \hat{\gamma}, \hat{\rho})$ is the baseline cumulative hazard function. Because of the relationship $H(t_i|\mathbf{z}_i, \boldsymbol{\theta}) = -\log S(t_i|\mathbf{z}_i, \boldsymbol{\theta})$, these residuals behave approximately like a censored sample from a standard exponential distribution (Lawless 2003). Then, a plot of $-\log \hat{S}[H(t_i|\mathbf{z}_i, \hat{\boldsymbol{\theta}})]$ versus $H(t_i|\mathbf{z}_i, \hat{\boldsymbol{\theta}})$ should be roughly a straight line with unit slope when the model is adequate, where $\hat{S}[H(t_i|\mathbf{z}_i, \hat{\boldsymbol{\theta}})]$ is the Kaplan-Meier estimate (Kaplan and Meier 1958) of $H(t_i|\mathbf{z}_i, \hat{\boldsymbol{\theta}})$. Note that for the MKumW, $H(t_i|\mathbf{z}_i, \hat{\boldsymbol{\theta}}) = -\hat{\rho}\log\{1 - [1 - \exp(-t_i^{\hat{\kappa}})]^{\hat{\gamma}}\}\exp(\mathbf{z}_i\hat{\boldsymbol{\beta}})$.

4.6.2 Proportionality Assumption

There are two widely used approaches to assess the PH assumption (1) statistical test, and (2) graphical diagnostics based on (a) scaled Schoenfeld residuals (Grambsch and Therneau 1994), and (b) time-dependent covariates (Lawless 2003). The Schoenfeld residuals are defined specifically for the Cox model, whereas the approach of using time-dependent covariates can be implemented in both the Cox and parametric PH models (Lawless 2003). We consider here time-dependent covariates to assess the PH assumption of the time-independent covariates under the MKumW model formulation. When assessing only one covariate, say z_1, the PH model (dropping the subscript i for simplicity) can be written as

$$h(t;\mathbf{z}) = h_0(t)\exp\{\mathbf{z}'\boldsymbol{\beta} + \beta_1^* z_1 g(t)\}, \tag{4.22}$$

where $g(t)$ is a known function of t and β_1^* is the regression coefficient associated with the time-dependent covariate $z_1 g(t)$. The PH assumption is then checked by assessing the significance of the term $z_1 g(t)$ in equation (4.22), i.e. $\mathrm{H}_0 : \beta_1^* = 0$. The Wald test or the likelihood ratio test can be used for this purpose. This approach can be extended for multiple covariates for which the model is

$$h(t;\mathbf{z}) = h_0(t)\exp\{\mathbf{z}'\boldsymbol{\beta} + \mathbf{z}^{*\prime}(t)\boldsymbol{\beta}^*\}, \tag{4.23}$$

where $\mathbf{z} = (z_1, z_2, \ldots, z_p)'$, $\mathbf{z}^*(t) = (z_1 g_1(t), z_2 g_2(t), \ldots, z_p g_p(t))'$, and $\boldsymbol{\beta}^* = (\beta_1^*, \beta_2^*, \ldots, \beta_p^*)'$. We do a test for assessing the PH assumption under $\mathrm{H}_0 : \beta_j^* = 0$. This test can be carried out under the assumption of the asymptotic normality of $\hat{\beta}_j^*$: $\hat{\beta}_j^*/\mathrm{se}(\hat{\beta}_j^*) \sim N(0,1)$. To assess all covariates simultaneously score test based on $\boldsymbol{\beta}^{*\prime}[\mathrm{cov}(\boldsymbol{\beta}^*)]^{-1}\boldsymbol{\beta}^* \sim \chi_p^2$ can be used under H_0.

The function $g_j(t)$ is commonly used to have a simple form. For example, $g_j(t)$ can be defined as an indicator function $I(t \geq t_0)$, which equals 1 if t is greater than or equal to a pre-specified value t_0 and 0 otherwise (Kleinbaum and Klein 2012). This approach can be handled using the maximum

likelihood method described above. Another widely used approach is to assume a continuous time-dependent covariates such as $g_j(t) = \log(t)$ or simply $g_j(t) = t$ (Kleinbaum and Klein 2012). Under this setting, $\beta_j(t)z_j = \beta_j + \beta_j^* g_j(t)$. Therefore, $\beta_j(t) = \beta_j + \beta_j^* g_j(t)$ is a time-dependent coefficient associated with the j^{th} covariate z_j. The Wald test for $H_0 : \beta_j^* = 0$ is then used to check whether this coefficient significantly varies with time. For estimation, modification of the maximum likelihood method is necessary to handle time-dependent coefficients. The log-likelihood contribution from individual i can be expressed as

$$\ell_i(\boldsymbol{\theta}) = \delta_i\{\log h_0(t_i) + \mathbf{z}'\boldsymbol{\beta} + \mathbf{z}^{*\prime}(t)\boldsymbol{\beta}^*\} - \exp\{\mathbf{z}'\boldsymbol{\beta}\} \int_o^{t_i} h_0(u) \exp\{\mathbf{z}^{*\prime}(u)\boldsymbol{\beta}^*\} du, \tag{4.24}$$

where the integral does not have an analytic solution with $h_0(u)$ of the form (4.9). In our implementation, the integral is approximated using a 15-point for numerical approximation (Davis and Rabinowitz 2007).

A graphical technique is also useful to assess the PH assumption of the covariates individually. For example, if $g_j(t) = \log t$ and $H_0 : \beta_j^* = 0$ is true, then a plot of time-dependent coefficient $\widehat{\beta_j(t)} = \hat{\beta}_j + \hat{\beta}_j^* \log t$ against t should closely approximate a horizontal line at $\hat{\beta}_j$.

4.7 Model Comparison

Selection of an appropriate approximation model is desirable to assign some preference to the alternatives. Graphical procedures, such as residual plots for assessing the best suitable model may involve subjective-approach when fits are reasonably close. Therefore, statistical tests or goodness of fit criteria is desirable to find "best" model. There are many analytical/statistical methods to combine information from competing models: one modern paradigm is the Akaike Information Criteria (AIC) introduced by Akaike (1974). The AIC is defined by

$$AIC = -2\ell + 2(p + k), \tag{4.25}$$

where ℓ is the log-likelihood, p is the number of covariates in the model and k is the number of parameters of the assumed probability distribution. When comparing two or more models, we prefer the one with the lowest AIC value. In this chapter, we will use Akaike information criterion (AIC) to compare the maximum likelihood fits of different models.

4.8 An Application to Pulmonary Exacerbation Data

Fuchs et al. (1994) (see also Lawless 2003) described a clinical trial to investigate the efficacy of daily administration of a recombinant form of the human enzyme DNase 1 in preventing pulmonary exacerbations. Exacerbation-free subjects were randomly assigned to either a new treatment (called rhDNase) or a placebo, and followed for approximately 169 days. There were 321 subjects in the treatment group and 324 in the placebo group. The day at which the first exacerbation period started was noted for each subject. There were 217 censored observations (i.e., no exacerbation occurred during the study period) in the rhDNase group, and 185 in the placebo group. Forced expiratory volume (fev) was also measured on each subject at randomization. The objective was to compare the two groups in terms of the avoidance of exacerbations. Khan (2017) analyzed these data using accelerated failure time models, and considered two covariates: trt = I(treatment = rhDNase) and fevc = fev – $\overline{\text{fev}}$, where $\overline{\text{fev}}$ is the mean fev across all subjects in the study. We consider here a PH model of the form $h(t;\mathbf{z}) = h_0(t)\exp(\beta_1\,\text{trt} + \beta_2\,\text{fevc})$ to analyze these data.

Table 4.1: Maximum likelihood fits of the MKumW, Weibull and Cox PH models to the pulmonary exacerbation data.

	MKumW PH		Weibull PH		Cox PH	
Parameter	Estimate	SE	Estimate	SE	Estimate	SE
β_1 (trt)	−0.388	0.130	−0.392	0.130	−0.383	0.130
β_2 (fevc)	−0.021	0.003	−0.021	0.003	−0.021	0.003
υ	0.616	1.213	−6.175	0.332		
τ	1.641	0.259	−0.082	0.059		
ξ	2.991	0.074				

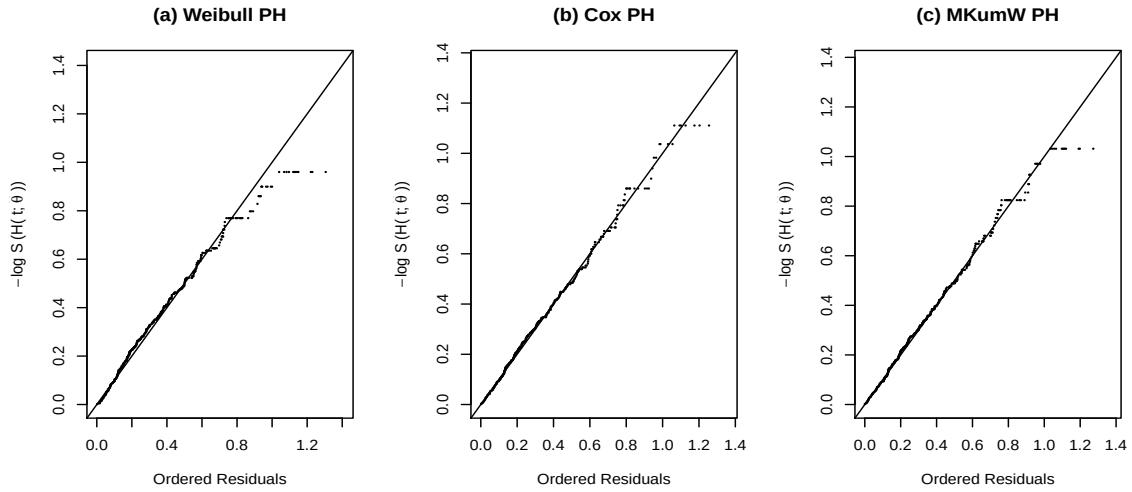

Figure 4.3: Diagnostic plots of the MKumW, Weibull and Cox PH models fitted to the pulmonary exacerbation data.

The maximum likelihood fits of the MKumW, Weibull and Cox PH models are summarized in Table 4.1. The likelihood ratio test of the Weibull as a submodel of MKumW suggests Weibull's inadequacy to describe the data ($\Lambda = 8.628$ on 1 df, with p-value < 0.01). The AIC goodness of fit criterion and residual plots (Figure 4.3) lead to the same conclusion: (a) AIC = 3261.615 and 3268.243 for the MKumW and Weibull PH fits, respectively, and (b) residuals of the MKumW PH fit lie closely to the unit-slope line, whereas residuals of the Weibull fit deviate slightly from the unit-slope line in the upper left corner. The MKumW PH and Weibull estimates of the regression coefficients are comparable (Table 4.1), though the distributional parameters vary considerably. Note that the distributional parameters of a parametric model play an important role in determining the adequacy of the overall fit (Khan 2017). The estimates of the MKumW shape parameters suggest unimodal hazard rates ($\hat{\kappa} = e^{-1.641} = 0.19$ with 95% confidence interval $(0.12, 0.32)$, and $\widehat{\kappa\gamma} = e^{-1.641+2.991} = 3.86$ with 95% confidence interval $(2.40, 6.20)$), whereas the estimate of the Weibull shape parameter indicates roughly a constant hazard function ($\hat{\kappa} = e^{0.082} = 1.09$ with 95% confidence interval $(0.97, 1.22)$, suggesting that κ is not significantly different from 1). This contradiction is due to the fact that the Weibull cannot handle nonmonotone hazard rates.

Table 4.2: Numerical results to check the PH assumption for each covariate in the Cox and MKumW fits to the pulmonary exacerbation data.

	MKumW PH			Cox PH		
Parameter	Estimate	SE	P-value	Estimate	SE	P-value
β_1 (trt)	−0.861	0.599	0.151	−0.978	0.656	0.14
β_2 (fevc)	−0.026	0.013	0.044	−0.030	0.014	0.04
β_1^* (trt × log t)	0.116	0.144	0.419	0.147	0.158	0.35
β_2^* (fevc × log t)	0.001	0.003	0.685	0.002	0.003	0.50
υ	−5.717	0.543	0.000			
τ	−0.005	0.106	0.963			
ξ	1.676	0.622	< 0.07			

The residual plots also suggest that the MKumW and Cox fits are comparable. To check the PH assumption, we take $g_j(t) = \log t$ and consider a model of the form $h(t;\mathbf{z}) = h_0(t)\exp\{\beta_1 \text{ trt} + \beta_2 \text{ fevc} + \beta_1^* (\text{trt} \times \log t) + \beta_2^* (\text{fevc} \times \log t)\}$ (see Section 4.6.2 for detail). We then fit two models separately: a PH model assuming MKumW baseline hazard (i.e., MKumW PH model), and a PH model assuming arbitrary baseline hazard (i.e., Cox PH model). Numerical results are summarized in Table 4.2. We see that the MKumW and the Cox PH fits are very close with respect to the estimates of the regression coefficients. Both the models suggest no evidence against the PH assumption (MKumW PH fit gives p-value = 0.419 and 0.685 for H_0: $\beta_1^* = 0$ and H_0: $\beta_2^* = 0$, respectively). Figure 4.4 shows the plots of $\hat{\beta}_j(t) = \hat{\beta}_j + \hat{\beta}_j^* \log t$ against time: solid lines (in gray) represent the MKumW PH fit, whereas the dashed lines (in gray) represent the Cox fit, overlaid with a smoothing spline fit of the scaled Schoenfeld residuals (solid lines in black) and a two-standard-error band around this fit (dashed lines in black). We see that the MKumW PH and Cox fits (i.e., $\hat{\beta}_j(t)$ lines) virtually coincide for each of the covariates. An approximately horizontal line for each $\hat{\beta}_j(t)$ indicates that both trt and fevc satisfy the PH assumption.

In summary, the above example demonstrates the following points of interest: (a) the proposed MKumW model fits the data very well, (b) the Weibull PH model fails to identify the unimodal shape of the hazard function, a quantity which is of fundamental interest in many medical applications, (c) the proposed MKumW PH model estimates the hazard function to be unimodal, and the overall MKumW fit appears superior compared to the Weibull PH fit, (d) the MKumW and the Cox

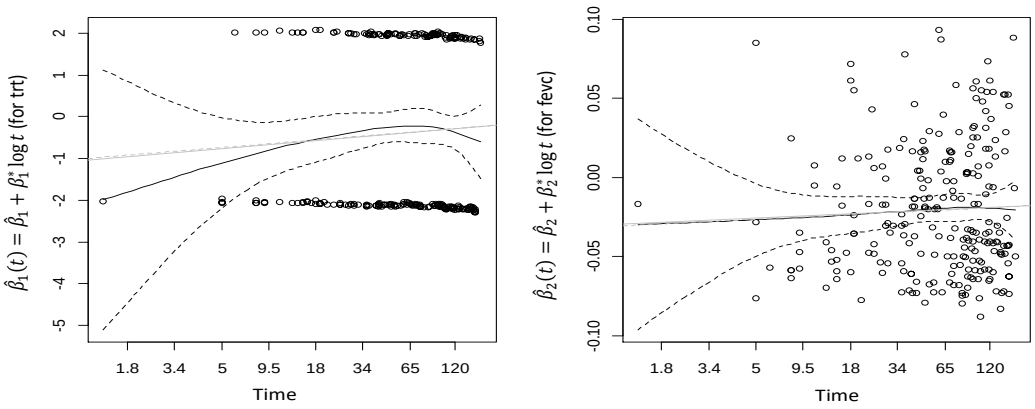

Figure 4.4: Diagnostic plots to check the PH assumption of each covariate in the Cox and MKumW fits to the pulmonary exacerbation data. For the Cox model, plots of scaled Schoenfeld residuals (a smoothing spline fit) against time are indicated by the solid lines (in black), and a two-standard-error band around the fit are indicated by the dashed lines (in black). Superimposed are the lines (in gray) for the time-dependent coefficients $\hat{\beta}_j(t)$; the solid lines (in gray) represent the MKumW fit, whereas the dashed lines (in gray) represent the Cox fit. These fits (i.e., $\hat{\beta}_j(t)$ lines) virtually coincide for each covariate.

PH fits are comparable, leading to the conclusion that the proposed model performs well (recall that the Cox PH model is robust against the distributional assumption of the survival time). These points also demonstrate that the MKumW model can be valuable in joint modeling of time-to-event and longitudinal data (see Chapter 5), as the use of the Cox PH in joint modeling usually leads to an underestimation of the standard errors of the estimates (Hsieh et al. 2006).

4.9 Simulation

Two covariates were considered in all simulations: one continuous covaraite (z_1) generated from the standard normal distribution, and one binary covariate (z_2) drawn from the Bernoulli(0.5) distribution. The regression coefficients corresponding to these two covaraites were chosen to be $\beta_1 = 0.5$ and $\beta_2 = -0.5$. To allow reasonable generalization, we considered four simulation scenarios: lifetime data exhibiting increasing (Scenario 1), decreasing (Scenario 2), unimodal (Scenario 3), and bathtub (Scenario 4) hazard shapes. A PH model of the form $h(t; \mathbf{z}) = h_0(t) \exp(0.5z_1 - 0.5z_2)$

was used to generate lifetime data, where $h_0(t)$ was modeled by the Weibull and the generalized Weibull (GW) (Mudholkar et al. 1996) hazard functions for Scenarios 1-2 and 3-4, respectively. Note that the baseline hazard function of the GW model is $h(t) = \frac{\kappa\rho t^{\kappa-1}}{1-\gamma\rho t^{\kappa}}$ for $\kappa, \rho > 0$ and γ is real; the hazard function is (a) increasing if $\kappa \geq 1$ and $\gamma \geq 0$, (b) decreasing if $\kappa \leq 1$ and $\gamma \leq 0$, (c) unimodal if $\kappa > 1$ and $\gamma < 0$, and (d) bathtub shaped if $\kappa < 1$ and $\gamma > 0$ (the model is nonregular (the support depends on some parameters) for increasing and bathtub-shaped hazard functions).

Our choice of the true models (i.e., Weibull and GW) was driven by two considerations: (a) model parsimony, and (b) the ability to accommodate monotone and nonmonotone hazard shapes (the Weibull accommodates only monotone hazard shapes, whereas the GW can handle nonmonotone hazards). To ensure that the conditions under which the simulation study conducted are as realistic as possible, the average fractions of censored data were taken to be around 20%, 35% and 50% for each scenario, representing light to heavy censoring cases. Note that the censored data were generated independently from the exponential distribution. Model parameter values and typical shapes of the hazard functions for Scenarios 1-4 are shown in Figure 4.5.

For each scenario, the MKumW model was used to analyze 500 simulated data sets, each of size $n = 100$. The estimates of the parameters were then averaged over the 500 sets, and the coverage probabilities of 95% confidence intervals (proportion of such intervals out of 500 that capture the truth) were calculated for β_1 and β_2. Note that the distributional parameters may differ considerably for different models, and are not directly comparable (Khan 2017). Under the current setting, the true values of the distributional parameters for the MKumW model are unknown, and therefore coverage probabilities for these parameters are not reported here. However, the estimates of the MKumW shape parameters are expected to correctly identify the true shape of a hazard function.

Numerical results are summarized in Table 4.3. We see that the MKumW model performs well with respect to the regression coefficients in all scenarios: the average of the estimates for each regression coefficient is close to the true parameter value, and the corresponding coverage

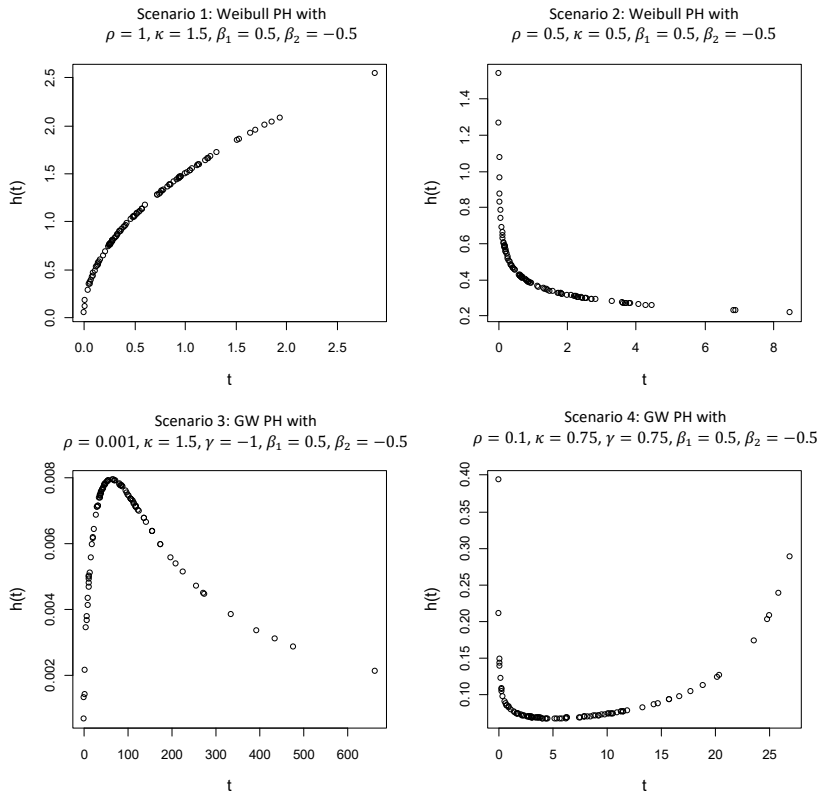

Figure 4.5: Typical hazard shapes for simulation scenarios 1-4: with one continuous covariate (z_1) and one binary covariate (z_2), lifetime data were generated from $h(t; \mathbf{z}) = h_0(t) \exp(0.5z_1 - 0.5z_2)$, where $h_0(t)$ is the Weibull and generalized Weibull (GW) hazard functions for Scenarios 1-2 and 3-4, respectively; the censored data were generated from the exponential distribution for all scenarios.

probabilities are all reasonably close to the nominal 0.95. On average, the proposed model also correctly identifies the shape of the hazard function. For example, $\hat{\kappa} = \exp(-1.470) = 0.23 < 1$ and $\hat{\kappa}\hat{\gamma} = \exp(-1.470 + 3.182) = 5.54 > 1$ for Scenario 3 with about 20% censored data, suggesting (on average) a unimodal hazard shape.

Note that the MKumW model is expected to perform well when the true model is Weibull, as the Weibull distribution is nested in the MKumW family of distributions. In contrast, the MKumW may lead to under coverage for the regression coefficients if the distributional assumption is not satisfied. For example, some other combinations of the GW parameters for Scenario 4 resulted in under coverage (coverage probability ≈ 0.90) for the regression coefficients. Thus, unlike the Cox

Table 4.3: Model performance using simulation study ($n = 100$) with one continuous covariate (z_1) and one binary (z_2) covariate (data simulated from a PH model of the form $h(t;\mathbf{z}) = h_0(t)\exp(0.5z_1 - 0.5z_2)$): table entries are average of 500 estimates of the MKumW parameters and coverage of 95% confidence intervals.

			Censoring ≈ 20%		Censoring ≈ 35%		Censoring ≈ 50%	
Scenario	True Model	Parameter	Mean	Coverage	Mean	Coverage	Mean	Coverage
Scenario 1	Weibull PH:	β_1	0.518	0.960	0.519	0.968	0.521	0.958
(Increasing	$\rho = 1$,	β_2	−0.513	0.946	−0.523	0.940	−0.533	0.944
hazard)	$\kappa = 1.5$,	τ	−0.377		−0.346		−0.361	
	$\beta_1 = 0.5$,	ξ	0.050		0.091		0.072	
	$\beta_2 = -0.5$	υ	0.139		0.242		0.253	
Scenario 2	Weibull PH:	β_1	0.516	0.954	0.519	0.962	0.517	0.966
(Decreasing	$\rho = 0.5$,	β_2	−0.512	0.950	−0.512	0.958	−0.514	0.938
hazard)	$\kappa = 0.5$,	τ	0.287		0.334		0.429	
	$\beta_1 = 0.5$,	ξ	−0.009		0.055		0.176	
	$\beta_2 = -0.5$	υ	−0.659		−0.556		−0.366	
Scenario 3	GW PH:	β_1	0.534	0.956	0.532	0.950	0.528	0.956
(Unimodal	$\rho = 0.001$,	β_2	−0.524	0.936	−0.520	0.934	−0.526	0.948
hazard)	$\kappa = 1.5$,	τ	1.470		1.530		1.462	
	$\gamma = -1$,	ξ	3.182		3.223		3.201	
	$\beta_1 = 0.5$,	υ	0.806		1.284		1.189	
	$\beta_2 = -0.5$							
Scenario 4	GW PH:	β_1	0.430	0.924	0.455	0.954	0.482	0.968
(Bathtub	$\rho = 0.1$,	β_2	−0.430	0.948	−0.454	0.950	−0.474	0.942
hazard)	$\kappa = 0.75$,	τ	−0.274		−0.216		−0.146	
	$\gamma = 0.75$,	ξ	−1.107		−0.915		−0.709	
	$\beta_1 = 0.5$,	υ	−3.304		−3.128		−2.944	
	$\beta_2 = -0.5$							

PH model, the MKumW is not robust as is any other parametric models. However, as highlighted in Section 4.1, a parametric model may lead to more efficient estimates than the Cox's model if the distributional assumption is satisfied.

4.10 Conclusion

PH models are widely used to analyze time-to-event data. Many studies have demonstrated that a parametric PH model can produce more efficient results than the semi-parametric Cox model under certain conditions (e.g. Efron 1977, Hsieh et al. 2006, Oakes 1977). However, only a few probability distributions are closed under the PH relationship. There is also a trade-off between

parsimony and flexibility among these models. In this Chapter, we propose a three-parameter probability distribution, which is closed under the PH relationship. The model is flexible in the sense that it accommodates both monotone (increasing and decreasing) and nonmonotone (unimodal and bathtub shape) hazard rates. We then formulate a PH regression model based on the proposed distribution, and develop large sample theory for statistical inference. A real data example demonstrates that the proposed model can outperform the Weibull PH model, which is the most commonly used parametric PH model for analyzing time-to-event data to date. The fit of the proposed model is also comparable to that of the Cox PH model. In summary, the proposed model has a wide range of applications because of its flexibility in modeling different types of hazard functions. A simulation study also reveals that the model can be valuable in adequately describing different types of time-to-event data.

Applications of PH models are commonly seen in recurrent event data analysis (Cook and Lawless 2007)) and joint modeling of longitudinal and time-to-event data (Rizopoulos 2012). In Chapter 5, we develop methods for recurrent event data analysis based on our MKumW distribution. There, we also propose a Bayesian approach for joint modeling using the MKumW model. Note that with respect to statistical efficiency, methods for joint modeling based on parametric response distributions are generally preferred, as opposed to a semi-parametric method like the Cox PH model. Based on this ground, the proposed model can be very useful in joint modeling applications.

CHAPTER 5

RECURRENT EVENT DATA ANALYSIS AND JOINT MODELING

In Chapter 3 and 4 we proposed two parametric PH models for a relatively simpler situation, where the event of interest is assumed to occur only once for a given subject (i.e., single-event survival data). However, more complicated situations may arise in practice. In this chapter, we consider two such commonly encountered problems in survival analysis, which require further extension of the standard PH model. More specifically, we propose methods for (1) recurrent event data analysis, where an individual can experience an event multiple times over follow-up (e.g., recurrent heart attacks of coronary patients); and (2) joint modeling for single-event survival data, where the event time distribution depends on a longitudinally measured internal covariate (e.g., prostate specific antigen (PSA) measurements may be obtained for patients following treatment for prostate cancer (longitudinally measured internal covariate), along with time to disease recurrence). The theoretical development for these two problems are based on our Modified Kumaraswamy Weibull (MKumW) model, which we consider here as it is more flexible than the generalized log-logistic model of Chapter 3.

In Section 5.1, we present the context and motivation of this study by highlighting the recurrent event and joint modeling problems in survival analysis. We then propose a framework for recurrent event data analysis based on our MKumW distribution, and demonstrate our method with an application to bladder cancer data (Section 5.2). The joint modeling problem is considered in Section 5.4, where we develop a Bayesian approach by linking our MKumW model for time-to-event data

to the linear mixed-effects model for longitudinal data; these two models are assumed to depend on a set of latent (unobserved) random effects. The proposed joint model is demonstrated with an application to AIDS data. There, we also compare the performance of our model with the Weibull joint model using residual analysis and deviance information criterion. We conclude in Section 5.5 by summarizing our findings.

5.1 Introduction

Recurrent event data analysis and joint modeling of longitudinal and time-to-event data are two very important areas of research in survival analysis, as many data from different applications (e.g., medical, clinical and biological studies, and engineering) fall under these setups. PH models offer an attractive modeling paradigm for these problems (Tsiatis and Davidian 2004, Rizopoulos 2012, Cook and Lawless 2007). The main focus of this chapter is to develop methods for recurrent event data analysis and joint modeling based on our flexible MKumW PH model.

Recurrent event data arise when the individuals under study may experience multiple events over time. In some settings, the events can be of the same type (e.g., recurrence of bladder cancer tumors), whereas in some other settings the events can be of different types (e.g., multiple sequelae due to diabetes mellitus). The former problem can be handled using a marginal model approach (Andersen and Gill 1982, Lin 1994, Wei et al. 1989), and the latter problem can be dealt with methods involving stratified Cox models. Nevertheless, the main objective of recurrent event data analysis is to assess the relationship of fixed and/or time-dependent covariates to event occurrence. In this study, we consider the setting where the events are of the same type. A major challenge in extending the PH models for recurrent event data is how to account for intra-individual correlation. Although there are several approaches, methods that incorporate intra-individual correlation through covariates or random effects have become popular. In particular, the Andersen-Gill model (Andersen and Gill 1982) considers a marginal modeling approach, assuming that the events occur

randomly so that the event occurrences in non-overlapping time intervals are statistically independent. Under this assumption, the Andersen-Gill model for recurrent event data can be developed based on the Poisson process formulation (Cook and Lawless 2007). The basic formulation of the Andersen-Gill model is similar to that of the Cox PH model, with the baseline intensity function assumed arbitrary. Note that the baseline intensity function can also be specified parametrically, leading to a fully parametric model (Cook and Lawless 2007). As highlighted in Chapters 3 and 4, there are several advantages of a parametric model, including the fact that the estimation of the baseline intensity function is quite straightforward. In Section 5.2, we propose a parametric recurrent event model, formulated using our MKumW distribution. Specifically, we consider the Poisson process formulation with the baseline intensity function modeled parametrically.

The other problem considered in this chapter is joint modeling. In many longitudinal studies, a longitudinal response is observed along with an observation of the time to the occurrence of an event; the event can be timed from the beginning of an observation period, resulting in survival or time-to-event data. A typical goal in such studies is to investigate the effects of the longitudinal response (internal covariate for the event process; for more details, see below) on the development of the event. It is also of particular interest to understand the within-individual trends of the longitudinal response. Some common issues encountered in such studies involve (a) measurement error in the longitudinal response, (b) missing information over time (longitudinal response measurements are usually collected only intermittently over time), and (c) censored observations in the event process. Failure to account these issues may lead to biased estimates of the regression parameters. The modern approach to analyze these types of data involves two separate models: a model that takes into account measurement error and missing data in the internal time-dependent covariate (longitudinal response) to estimate its true values (longitudinal model), and another model that uses these estimated values to quantify the association between this covariate and the time to the occurrence of the event (time-to-event model). The motivating idea behind the joint modeling techniques is to couple the time-to-event model with the longitudinal model through shared random effects (Tsiatis

and Davidian 2004). This technique allows the underlying relationship between the event process and the longitudinal process to be acknowledged explicitly.

The motivation of joint modeling originated from clinical trial studies, where the internal covariate is a surrogate marker. Considering a treatment/disease process, the surrogacy can be explained using the following three conditions "(I) treatment must have an effect on the time-to-event; (II) treatment must have effect on the marker; and (III) effect of treatment should manifest through the marker, i.e., the risk of the event given a specific marker trajectory should be independent of treatment." For example, longitudinal CD4 count may be considered as a surrogate marker for time to progression to AIDS or death, and joint modeling may be used to quantify the association between CD4 cells counts and time until death (Guo and Carlin 2004).

The standard approach for joint modeling is to consider a linear mixed-effects model for the internal covariate and a PH model for the association analysis (Guo and Carlin 2004, Henderson et al. 2000, Williamson et al. 2008, Wu et al. 2012). Although the Cox PH model is appealing to analyze standard survival data mainly due to its robustness property, the use of the Cox PH in joint modeling usually leads to an underestimation of the standard errors of the parameter estimates (Hsieh et al. 2006, Rizopoulos 2012). Therefore, most methods for joint modeling are based on parametric response distributions. For example, Guo and Carlin (2004) considered the Weibull model for joint analysis of longitudinal CD4 count and time to death of the AIDS patients. Since the Weibull distribution accommodates only monotone failure rates, it is desirable to consider a more flexible time-to-event model in joint analysis. With this motivation, we propose a joint modeling framework based on our MKumW distribution.

Fitting a joint model using the likelihood method involves computationally intensive methods. It requires evaluation of multiple integrals with respect to time and random effects, combined with the requirement of numerical optimization for maximum likelihood estimation (Rizopoulos 2012). Although the integral with respect to time can be well approximated using the Gauss-Kronrod rule (Press et al. 2007), the integrals with respect to the random effects are computationally demanding

to approximate, especially when the dimensionality of the random effects increases. Therefore, we propose a Bayesian approach for joint modeling, implemented through the Markov Chain Monte Carlo (MCMC) algorithm. For computation, we use the R package R2WinBUGS (Sturtz et al. 2005) to run WinBUGS (Lunn et al. 2000) from R Core Team (2018). Note that WinBUGS is a statistical software for Bayesian analysis using MCMC methods.

5.2 Recurrent Event Data Analysis

The general approach of the Poisson process formulation for modeling recurrent event data is described in Cook and Lawless (2007). For a Poisson process, it is assumed that the events occur randomly in such a way that the numbers of events in non-overlapping time intervals are statistically independent. Under this assumption, the Poisson intensity function for modeling a recurrent event process can be expressed as in Cook and Lawless (2007)

$$h(t;\mathbf{z}) = h_0(t)\exp[\mathbf{z}'(t)\boldsymbol{\beta}], \tag{5.1}$$

where $\mathbf{z}(t)$ is a vector of external, possibly time-dependent, covariates (i.e., values are determined independently of the recurrent event process), $\boldsymbol{\beta}$ is a vector of regression parameters of the same length as $\mathbf{z}(t)$, and $h_0(t)$ is the baseline intensity function. Given the covariates, the inter-individual variation in event occurrence is accounted for by the Poisson process. Note that a parametric specification for $h_0(t)$ leads to a fully parametric model.

Suppose that individual i ($i = 1, 2, \ldots, n$) is under observation from time $t_{i0} = 0$ to $t_{in_i} = t_i$, and event occurrences and covariate information are recorded at times $t_{i1} < t_{i2} < \ldots < t_{in_i}$. Let d_{ik} and z_{ijk} denote the event status (1 if an event occurs, and 0 otherwise) and the value of the j^{th} covariate in the interval $(t_{i,k-1}, t_{ik}]$, respectively, where $i = 1, 2, \ldots, n$, $j = 1, 2, \ldots, p$. Letting $\mathbf{z}_{ik}$ be the $p \times 1$ vector of covariates for individual i in the interval $(t_{i,k-1}, t_{ik}]$, and $\boldsymbol{\beta} = (\beta_1, \beta_2, \ldots, \beta_p)'$ the corresponding vector of regression coefficients, the log-likelihood contribution from individual

i can be expressed as

$$\ell_i(\boldsymbol{\theta}) = \sum_{k=1}^{n_i} d_{ik}\{\log h_0(t_{ik}) + \mathbf{z}_{ik}'\boldsymbol{\beta}\} - \int_0^{t_i} h_0(u)\exp\{\mathbf{z}_i'(u)\boldsymbol{\beta}\}du,$$

which can be rewritten (under the assumption that z_{ijk} is fixed over the interval $(t_{i,k-1}, t_{ik}]$ for all j and k as

$$\ell_i(\boldsymbol{\theta}) = \sum_{k=1}^{n_i} d_{ik}\{\log h_0(t_{ik}) + \mathbf{z}_{ik}'\boldsymbol{\beta}\} - \sum_{k=1}^{n_i}\int_{t_{i,k-1}}^{t_{ik}} h_0(u)\exp(\mathbf{z}_{ik}'\boldsymbol{\beta})du. \tag{5.2}$$

Using the MKumW PH hazard function for $h_0(\cdot)$, (5.2) can be expressed as

$$\begin{aligned}\ell_i(\boldsymbol{\theta}) = {} & m_i\log\kappa + m_i\log\gamma + m_i\log\rho + (\gamma-1)\sum_{k=1}^{n_i} d_{ik}\log(1-a_{ik}) \\ & - \sum_{k=1}^{n_i} d_{ik}\log[1-(1-a_{ik})^{\gamma}] + (\kappa-1)\sum_{k=1}^{n_i} d_{ik}\log t_{ik} - \sum_{k=1}^{n_i} d_{ik}t_{ik}^{\kappa} \\ & + \sum_{k=1}^{n_i} d_{ik}\mathbf{z}_{ik}'\boldsymbol{\beta} + \rho\sum_{k=1}^{n_i}\log\Big\{\frac{1-(1-a_{ik})^{\gamma}}{1-(1-a_{i,k-1})^{\gamma}}\Big\}\exp(\mathbf{z}_{ik}'\boldsymbol{\beta}),\end{aligned} \tag{5.3}$$

where $m_i = \sum_{k=1}^{n_i} d_{ik}$ and $a_{ik} = \exp(-t_{ik}^{\kappa})$. Maximizing $\ell(\boldsymbol{\theta}) = \sum_{i=1}^{n}\ell_i(\boldsymbol{\theta})$ then yields the maximum likelihood estimator $\hat{\boldsymbol{\theta}}$, which under mild regularity conditions has the usual asymptotic properties (Cook and Lawless 2007). For statistical inference, the observed information matrix can be computed using numerical differentiation as mentioned in Section 4.5.1. The goodness of fit techniques described in Sections 4.5.2 and 4.6 can also be used in the recurrent event setting. Note that the semi-parametric model with $h_0(t)$ an arbitrary positive-valued function is called the Andersen-Gill model (Andersen and Gill 1982) model, for which statistical inference is carried out using the partial likelihood approach.

5.3 Model Assessment

The proportionality assumption can be tested using the time-dependent covariate approach as described in Section 4.6.2. More specifically, letting $\mathbf{z}_{ik}^*(t) = (z_{i1k}g(t), z_{i2k}g(t), \ldots, z_{ipk}g(t))'$ with $g(t)$

a known function of time t, the log-likelihood contribution from individual i can be written as

$$\begin{aligned}\ell_i(\boldsymbol{\theta}) &= \sum_{k=1}^{n_i} d_{ik}\{\log h_0(t_{ik}) + \mathbf{z}_{ik}'\boldsymbol{\beta} + \mathbf{z}_{ik}^{*'}\boldsymbol{\beta}^*\} - \int_0^{t_i} h_0(u)\exp[\mathbf{z}_{ik}'\boldsymbol{\beta} + \mathbf{z}_{ik}^{*'}(u)\boldsymbol{\beta}^*]du \\ &= \sum_{k=1}^{n_i} d_{ik}\{\log h_0(t_{ik}) + \mathbf{z}_{ik}'\boldsymbol{\beta} + \mathbf{z}_{ik}^{*'}\boldsymbol{\beta}^*\} - \sum_{k=1}^{n_i}\exp(\mathbf{z}_{ik}'\boldsymbol{\beta})\int_{t_{i,k-1}}^{t_{i,k}} h_0(u)\exp[\mathbf{z}_{ik}^{*'}(u)\boldsymbol{\beta}^*]du, \qquad (5.4)\end{aligned}$$

where $\mathbf{z}_{ik}^* = (z_{i1k}g(t_{ik}), z_{i2k}g(t_{ik}), \ldots, z_{ipk}g(t_{ik}))'$ and $\boldsymbol{\beta}^*$ is the parameter vector associated with $\mathbf{z}_{ik}^*$. Note that the integrals in (5.4) can be well approximated by the Gauss-Kronrod rule (Press et al. 2007). We can then test the proportionality assumption for specific covariates (i.e., $H_0 : \beta_j^* = 0$ for $j = 1, 2, \ldots, p$) using the Wald statistic $\hat{\beta}_j^*/SE(\hat{\beta}_j^*) \sim N(0, 1)$.

Graphical assessment of a recurrent event model fit can be carried out using residual analysis. Let

$$E_{ik} = \int_{t_{i,k-1}}^{t_{ik}} h_i(u; \mathbf{z})du \qquad (5.5)$$

for $k = 1, 2, \ldots, n_i$. Then, the quantities

$$R_{ik} = E_{i1} + E_{i2} + \ldots + E_{ik} \qquad (5.6)$$

can be viewed as the occurrence times of events in a homogeneous Poisson process (i.e., a transformed time scale). An estimate of E_{ik} can be obtained by replacing $h_i(u; \mathbf{z})$ with the maximum likelihood estimate $\hat{h}_i(u; \mathbf{z})$. For large samples, the $\hat{E}_{ik}$ should behave like standard exponential random variables if the specification of the model is correct (Cook and Lawless 2007). Thus, a plot of the Nelson-Aalen estimate of the transformed times $\hat{R}_{ik}$ against $\hat{R}_{ik}$ should be roughly linear with slope one if the proposed model is adequate.

Although graphical approach such as a comparison of residual plots can provide useful information with regard to comparison of two or more model fits, it may involve subjective decision-making when it comes to preferring one model over the others. This is especially true when the fits are fairly close. A more objective approach involves a comparison of the AIC values or the likelihood ratio test if one model is nested within another model. In this study, we consider all

these techniques to compare the performance of our model with that of the Weibull recurrent event model.

5.3.1 An Application to Bladder Cancer Data

Byar (1980) described a clinical trial for patients with bladder cancer (data are available in the R package "survival" (Therneau 2015)), where there were 85 subjects with bladder tumors at the time they entered the trial. These tumors were first removed, and then the patients were randomly assigned to receive either thiotepa treatment or placebo, resulted in 38 and 47 patients in the two groups, respectively. The number of initial tumors at the beginning of the study, and the diameter (in cm) of the largest such tumor, were recorded as covariates. The data in the R package "survival" also contain the months from the beginning of the study to each follow-up inspection, up to four recurrences of tumors, and an event indicator with 1 for recurrence of tumor and 0 for censored. We consider a model of the form $h(t;\mathbf{z}) = h_0(t)\exp(\beta_1 \text{ rx} + \beta_2 \text{ number} + \beta_3 \text{ size})$, where rx = I(treatment = thiotepa), number = number of initial tumors at the beginning of the study, and size = size (in cm) of the largest initial tumour.

Table 5.1: Maximum likelihood fits of the MKumW PH, Weibull and Andersen-Gill PH models to the bladder cancer data.

	MKumW PH		Weibull PH		Andersen-Gill	
Parameter	Estimate	SE	Estimate	SE	Estimate	SE
β_1 (rx)	−0.495	0.200	−0.510	0.199	−0.465	0.200
β_2 (number)	0.184	0.047	0.188	0.047	0.175	0.047
β_3 (size)	−0.042	0.069	−0.044	0.068	−0.044	0.069
υ	−2.329	0.628	−3.089	0.368		
τ	0.261	0.190	0.042	0.087		
ξ	1.349	0.430				

The maximum likelihood fits of the MKumW, Weibull and Andersen-Gill (i.e., arbitrary positive-valued baseline hazard) models are summarized in Table 5.1, and the residual plots are displayed in Figure 5.1 (see Appendix A.5 for R codes). The estimates of the regression coefficients are comparable for the three models. The residual plots also suggest no obvious preference of one model over

the other. However, the likelihood ratio test suggests that the MKumW provides slightly better fit than that of the Weibull model ($\Lambda = 5.358$ on 1 df, with p-value = 0.02). To supplement our overall assessment, we also compare the AIC values: AIC = 903.79, 907.15 and 905.96 for the MKumW, Weibull and Andersen-Gill models. The AIC values indicate the superiority of MKumW over the Weibull fit. Note that the AIC for the Andersen-Gill model is calculated based on the partial likelihood function, and therefore it is not directly comparable with that of a parametric model (AIC for a parametric model is calculated based on the full likelihood function). Nevertheless, the MKumW model fit gives a smaller AIC value compared to that of the Andersen-Gill model.

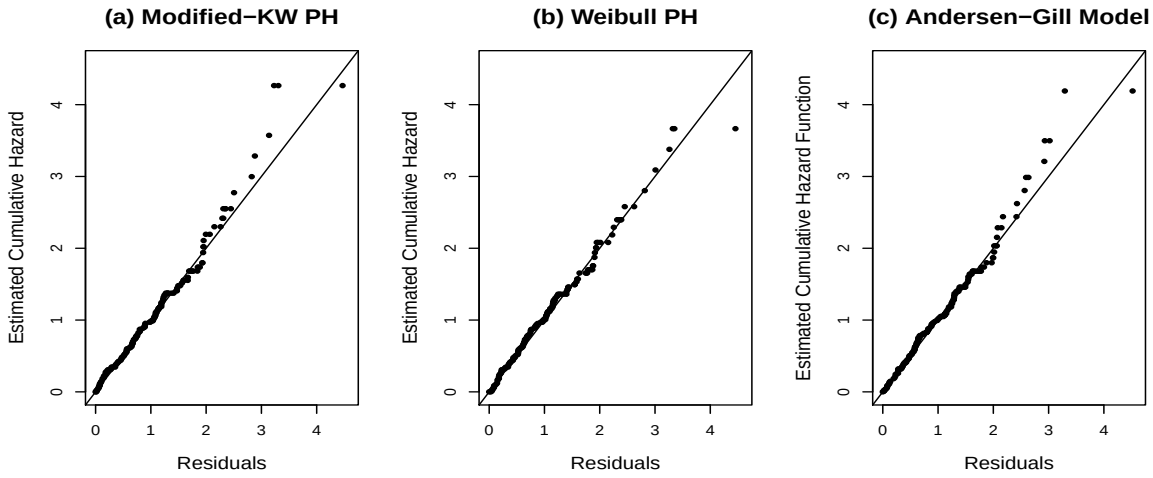

Figure 5.1: Residual plots of the MKumW, Weibull and Andersen-Gill PH models fitted to the bladder cancer data.

The estimates of the MKumW shape parameters indicate increasing hazard rates: $\hat{\kappa} = e^{-0.261} = 0.78$ with 95% confidence interval $(0.53, 1.12)$, suggesting that κ is not significantly different from 1, whereas $\widehat{\kappa\gamma} = e^{-0.261+1.349} = 2.97$ with 95% confidence interval $(1.55, 5.68)$, suggesting that $\kappa\gamma$ is significantly greater than 1. With respect to the scientific context of this problem, the hazard of tumor recurrence is indeed expected to increase over time. Note that the estimate of the Weibull shape parameter indicates roughly a constant hazard function ($\hat{\kappa} = e^{-0.042} = 0.96$ with 95% confidence interval $(0.81, 1.14)$).

Table 5.2: Numerical results to check the PH assumption for each covariate in the Andersen-Gill and MKumW PH fits to the bladder cancer data.

	MKumW PH			Andersen-Gill		
Parameter	Estimate	SE	P-value	Estimate	SE	P-value
β_{11} (tx)	−0.212	0.515	0.680	−0.295	0.540	0.59
β_{12} (number)	0.148	0.133	0.265	0.172	0.128	0.18
β_{13} (size)	0.158	0.169	0.349	0.181	0.172	0.29
β_{21} (tx $\times \log t$)	−0.112	0.198	0.570	−0.070	0.208	0.74
β_{22} (number $\times \log t$)	0.015	0.052	0.770	0.002	0.051	0.97
β_{23} (size $\times \log t$)	−0.082	0.066	0.215	−0.093	0.068	0.17
υ	1.836	2.603	0.481	–		-
τ	−1.521	0.561	0.007	–		-
ξ	2.561	0.451	0.000	–		-

The MKumW and Andersen-Gill fits using time-dependent covariates (to asses the PH assumption) are summarized in Table 5.2. Non-significant p-values for the time-dependent covariates suggest that the PH assumption holds for each covariate. The plots of the time-dependent coefficients $\hat{\beta}_j(t)$ (Figure 5.2) also support this fact (a plot of the time-dependent coefficient $\hat{\beta}_j(t)$ against t should closely approximate a horizontal line at $\hat{\beta}$ if the PH assumption holds).

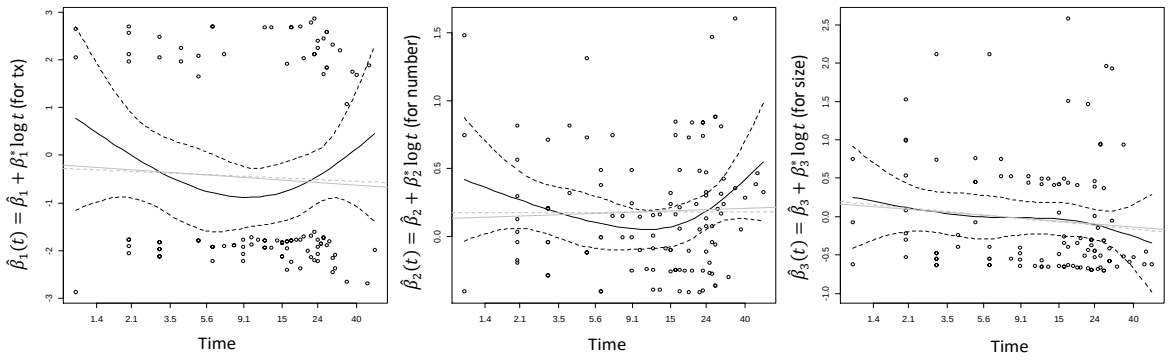

Figure 5.2: Diagnostic plots to check the PH assumption of each covariate in the Andersen-Gill and MKumW fits to the bladder cancer data. For the Andersen-Gill model, plots of scaled Schoenfeld residuals (a smoothing spline fit) against time are indicated by the solid lines (in black), and a two-standard-error band around the fit are indicated by the dashed lines (in black). Superimposed are the lines (in gray) for the time-dependent coefficients $\hat{\beta}_j(t)$; the solid lines (in gray) represent the MKumW PH fit, whereas the dashed lines (in gray) represent the Andersen-Gill fit.

5.4 Joint Modeling of Time-to-Event and Longitudinal Data

The general framework of the maximum likelihood method is described in several articles and books (e.g., Rizopoulos 2012, Wu et al. 2012). A comprehensive description of joint modeling framework is given in Chapter 2. Here in this Section 5.4.1, we will develop joint model of longitudinal and time-to-event data based on the MKumW PH model in Bayesian framework and implemented via Markove chain Monte Carlo (MCMC) method.

5.4.1 Model Description

For model formulation, we adopt the approach proposed by Henderson et al. (2000). Let there be n individuals with lifetimes denoted by $T_1, T_2, \ldots, T_n$. Assuming that the data are subject to right censoring, we observe $t_i = \min(T_i, C_i)$ for individual i ($i = 1, 2, \ldots, n$), where $C_i > 0$ corresponds to a potential censoring time and $\delta_i = \mathrm{I}(T_i \leq C_i)$ denotes the censoring indicator that equals 1 if $T_i \leq C_i$ and 0 otherwise. We consider the case of a single (internal) covariate that is measured over time. Specifically, we assume that individual i provides a set of longitudinal quantitative measurements $\{y_{ij} : j = 1, 2, \ldots, n_i\}$ at times $\{s_{ij} : j = 1, 2, \ldots, n_i\}$, where $s_{ij} \leq t_i$. Therefore, the observed data for individual i consist of $\{t_i, \delta_i, \mathbf{y}_i, \mathbf{s}_i\}$, where $\mathbf{y}_i = (y_{ij} : s_{ij} \leq t_i)$ and $\mathbf{s}_i = (s_{ij} : s_{ij} \leq t_i)$.

The above setup leads to two related processes: a time-to-event process to quantify the association between the internal covariate and the risk of an event, and a longitudinal process for the internal covariate to take into account measurement errors, missing data and subject-specific random effects. The joint distribution of longitudinal measurements and event times is modeled via a bivariate latent process $(U_i(t), V_i(t))$ (see below). We assume that this latent process is driven by two sub-models, as described below.

5.4.2 Measurement Model

We model the internal covariate (longitudinal response) y_{ij} at time s_{ij} by the relationship

$$y_{ij} = \mu_i(s_{ij}) + U_i(s_{ij}) + \epsilon_{ij}, \tag{5.7}$$

where $\mu_i(s_{ij})$ is the mean response, $U_i(s_{ij})$ incorporates subject-specific random effects, and $\epsilon_{ij} \sim N(0, \sigma^2)$ is a sequence of mutually independent measurement errors. We assume that the mean response at time s can be described by a linear model

$$\mu_i(s) = \mathbf{x}_i'(s)\boldsymbol{\alpha} \tag{5.8}$$

where $\mathbf{x}_i(s)$ is a vector of possibly time-dependent covariates and $\boldsymbol{\alpha}$ is a corresponding vector of regression coefficients. For $U_i(s)$, we assume a linear random effects model

$$U_i(s) = \mathbf{w}_i'(s)\mathbf{b}_i, \tag{5.9}$$

where $\mathbf{w}_i(s)$ is a vector of covariates (which may be a subset of $\mathbf{x}_i(s)$) and $\mathbf{b}_i \sim N(\mathbf{0}, \Sigma)$ is a corresponding vector of random effects. mean response $\mu_i(s)$ (see Equation (5.7)), $U_i(s)$ can be viewed as the true trajectory for individual i (Guo and Carlin 2004). Thus, it is assumed that y_{ij} contributes to the time-to-event model only through the random effects $U_i(s)$.

5.4.3 Time-to-Event Model

Assuming a PH model, the hazard function for the time-to-event process can be expressed as

$$h(t_i) = h_0(t_i; \boldsymbol{\zeta}) \exp\{\mathbf{z}_i'\boldsymbol{\beta} + V_i(t)\}, \tag{5.10}$$

where $h_0(t; \boldsymbol{\zeta})$ is the MKumW baseline hazard function with $\boldsymbol{\zeta} = (\kappa, \gamma, \rho)'$, $\mathbf{z}_i$ is a vector of baseline covariates, $\boldsymbol{\beta}$ is the corresponding vector of regression coefficients, and $V_i(t)$ denotes the true and

unobserved value of the longitudinal response that has a form similar to $U_i(t)$; see below.

5.4.4 Association Structure

Dependence between the measurement and the time-to-event sub-models is captured through the association structure, which can be specified in a number of ways. In our implementation, $V_i(t)$ is specified as $V_i(t) = \phi U_i(t)$ so that ϕ measures the association induced by the fitted longitudinal value at the event time $U_i(t)$.

5.4.5 Hierarchical Formulation of the Model

We propose a hierarchical formulation of the model for Bayesian inference. Given $\boldsymbol{\alpha}$ and $\mathbf{b}_i$, the log-likelihood contribution for individual i with the MKumW baseline hazard function can be expressed as

$$
\begin{aligned}
\ell_i &= \delta_i \log[h_0(t_i;\boldsymbol{\zeta}) \exp\{\mathbf{z}_i'\boldsymbol{\beta} + \phi U_i(t_i)\}] - \int_0^{t_i} h_0(s;\boldsymbol{\zeta}) \exp\{\mathbf{z}_i'\boldsymbol{\beta} + \phi U_i(s)\} ds \\
&= \delta_i \log[h_0(t_i;\boldsymbol{\zeta}) \exp\{\mathbf{z}_i'\boldsymbol{\beta} + \phi \mathbf{w}_i'(t_i)\mathbf{b}_i\}] - \int_0^{t_i} h_0(s;\boldsymbol{\zeta}) \exp\{\mathbf{z}_i'\boldsymbol{\beta} + \phi \mathbf{w}_i'(s)\mathbf{b}_i\} ds \\
&= \delta_i [\log\kappa + \log\gamma + \log\rho + (\kappa-1)\log t_i - t_i^{\kappa} + (\gamma-1)\log(1-e^{-t_i^{\kappa}}) \\
&\quad - \log\{1-(1-e^{-t_i^{\kappa}})^{\gamma}\} + \mathbf{z}_i'\boldsymbol{\beta} + \phi \mathbf{w}_i'(t_i)\mathbf{b}_i] \\
&\quad - \kappa\gamma\rho e^{\mathbf{z}_i'\boldsymbol{\beta}} \int_0^{t_i} \frac{(1-e^{-s^{\kappa}})^{\gamma-1} e^{-s^{\kappa}} s^{\kappa-1}}{1-(1-e^{-s^{\kappa}})^{\gamma}} e^{\phi \mathbf{w}_i'(s)\mathbf{b}_i} ds, \qquad (5.11)
\end{aligned}
$$

where the integral in (5.11) has no analytic solution. In our implementation, we use the 5-point Gauss-Legendre rule (Abbott 2005) to evaluate the integral numerically (see Appendix A.6 for a description of the Gauss-Legendre rule). Note that Equation (5.11) is not the full log-likelihood function of the frequentist approach; the likelihood function for the frequentist approach is obtained

by integrating out the random effects $\mathbf{b}_i$. This step is avoided in a Bayesian approach by expressing the model hierarchically (see below), where we use ℓ_i of Equation (5.11) to obtain likelihood contribution given $\mathbf{b}_i$, with $\mathbf{b}_i$ assumed to follow a known distribution (we use a multivariate normal distribution for $\mathbf{b}_i$, the most common choice in longitudinal data analyses (Fitzmaurice et al. 2011)).

To formulate a hierarchical model, we use "zeros trick" as described in the WinBUGS manual (Spiegelhalter et al. 2003): the likelihood contribution of a Poisson(λ) observation of zero is $\exp(-\lambda)$; if we set $\lambda_i = -\ell_i$ with observed data a vector of 0's, then we get the correct likelihood contributions. Using this technique, the joint model can be formulated as follows:

$$\left.\begin{array}{c} \left[y_{ij}|\boldsymbol{\alpha}, \mathbf{b}_i\right] \sim \mathrm{N}(\mathbf{x}_{ij}'\boldsymbol{\alpha} + \mathbf{w}_{ij}'\mathbf{b}_i, \sigma^2), \\ [0|\zeta, \beta, \mathbf{b}_i, \phi] \sim \mathrm{Poisson}(-\ell_i), \\ [\mathbf{b}_i|\Sigma] \sim \mathrm{MVN}(\mathbf{0}, \Sigma), \\ [\boldsymbol{\alpha}|\mathbf{m}_\alpha, \mathbb{M}_\alpha] \sim \mathrm{MVN}(\mathbf{m}_\alpha, \mathbb{M}_\alpha), \ \left[\boldsymbol{\beta}|\mathbf{m}_\beta, \mathbb{M}_\beta\right] \sim \mathrm{MVN}(\mathbf{m}_\beta, \mathbb{M}_\beta), \\ [\phi|a_0, a_1] \sim \mathrm{N}(a_0, a_1), \ \left[\kappa^{-1}|b_0, b_1\right] \sim \mathrm{Gamma}(b_0, b_1), \\ \left[\gamma^{-1}|c_0, c_1\right] \sim \mathrm{Gamma}(c_0, c_1), \ [\rho|d_0, d_1] \sim \mathrm{Gamma}(d_0, d_1), \\ \left[\sigma^{-2}|e_0, e_1\right] \sim \mathrm{Gamma}(e_0, e_1), \left[\Sigma^{-1}|\nu, \mathbb{R}\right] \sim \mathrm{Wishart}(\nu, \mathbb{R}) \end{array}\right\}, \tag{5.12}$$

where $\mathbf{x}_{ij}$ is a vector of covariates for individual i at time s_{ij}, $\mathbf{w}_{ij}$ is a vector of covariates corresponding to the random effects $\mathbf{b}_i$ for individual i at time s_{ij}, and a_0, a_1, b_0, b_1, c_0, c_1, d_0, d_1, e_0, e_1, ν and $\mathbb{R}$ are hyperprior parameters, all are assumed known. Since regression coefficients can take any value in the entire real line, we use normal priors for $\boldsymbol{\alpha}$, ϕ and $\boldsymbol{\beta}$. We choose gamma priors for κ, γ and ρ , as these are positive-valued random variables. Note that gamma priors are widely used in Bayesian literature for positive-valued random variables (Gelman et al. 2013). For σ^{-2} and Σ^{-1}, we consider conjugate priors gamma and Wishart, respectively.

For the gamma and Wishart distributions, we consider the parameterizations implemented in

the WinBUGS software (Lunn et al. 2000, Spiegelhalter et al. 2003). For example, b_0 and b_1 are the shape and the inverse-scale parameters of the Gamma(b_0, b_1) distribution, respectively, and ν and $\mathbb{R}$ are the degrees of freedom and the inverse-scale matrix of the Wishart distribution Wishart$(\nu, \mathbb{R})$, respectively. In our implementation, we choose hyperprior values that lead to fairly vague, minimally informative priors. In particular, we choose $\mathbf{m}_\alpha = \mathbf{0}$, $\mathbf{m}_\beta = \mathbf{0}$, diagonal matrices for $\mathbb{M}_\alpha$ and $\mathbb{M}_\beta$ with all the diagonal elements equal to 100000, $a_0 = 0$, $a_1 = 10000$, $b_0 = b_1 = c_0 = c_1 = d_0 = d_1 = 0.01$, $e_0 = e_1 = 0.1$, $\nu =$ (dimension of Σ) + 1, and a diagonal matrix for $\mathbb{R}$ with all the diagonal elements equal to 0.1.

5.4.6 Bayesian Inference

Bayesian inference is based on the posterior distribution of the parameters given the data: the posterior density of a parameter describes its behavior over a range of values (the support of the parameter space), whereas the posterior mean or median is considered as the point estimate of the parameter. A $100(1-2p)\%$ Bayesian credible interval for a parameter is $[c_p, c_{1-p}]$, where c_p and c_{1-p} are estimated as the p^{th} and $(1-p)^{th}$ quantiles of the posterior distribution of the parameter, respectively.

We consider MCMC methods for Bayesian inference (Givens and Hoeting 2005), where one first needs to construct one or more Markov chains that necessarily converge to a stationary distribution. The marginal posterior density is then estimated using the kernel density estimation, and the posterior mean, median, standard deviation and other summaries are approximated by their sample equivalents in the MCMC output.

There are several techniques to construct Markov chains, including Metropolis-Hastings algorithm, Gibbs sampling and importance sampling (Givens and Hoeting 2005). We consider the Gibbs sampling technique in our implementation, where the parameter vector is partitioned into a number of components of possibly differing dimensions, and then update each of these components one by one. For a particular component, an instance of a Markov chain is generated from its

full conditional distribution (i.e., the distribution of the component of interest conditioning on all the remaining components).

For MCMC analysis, the usual practice is to run two or more Markov chains with the hope that at least one of these will explore all the features of the target distribution. It may take some iterations for the chains to enter into the high probability region where they are more representative of the target distribution. Therefore, the first few iterations of the chains called burn-in are usually discarded. It is also a good practice for the inference to be based on every l^{th} iteration of a chain (thinning), with l set to some value high enough that successive draws are approximately independent (Gelman 1995). Ideally, we would like good mixing of the chains, meaning that the chains fully explore the support and shape of the target distribution.

A formal statistical tool to check mixing and convergence of the chains is the Gelman-Rubin statistic R (Gelman and Rubin 1992), developed based on a comparison of the within-chain and between-chain variabilities. Values of R substantially above 1 indicate lack of convergence. There are also graphical tools to assess the mixing property and convergence of the chains. The most widely used graphical technique is the trace plot, where one plots the realization of the chains versus the iteration number. A clear trend in the trace plot indicates that stationarity has not achieved.

The most widely used tool for model comparison in Bayesian analysis is the deviance information criterion (DIC, Spiegelhalter et al. 2002). This criterion is defined based on two components: goodness of fit and model complexity. The deviance statistic $D(\boldsymbol{\theta}) = -2 \log L(\text{data}|\boldsymbol{\theta})$ is used as a measure of goodness of fit, where $\boldsymbol{\theta}$ denotes all the model parameters collectively; a point estimate of the deviance can be obtained by substituting $\bar{\boldsymbol{\theta}}$ (posterior mean of $\boldsymbol{\theta}$) in $D(\boldsymbol{\theta})$, that is, $D(\bar{\boldsymbol{\theta}})$. Spiegelhalter et al. (2002) proposed a measure of complexity as the posterior mean deviance minus deviance evaluated at the posterior mean of the parameters, that is, $\bar{D} - D(\bar{\boldsymbol{\theta}})$, where $\bar{D}$ is the posterior mean of the deviance. This is also a measure of the effective number of parameters in the model. The DIC is then defined analogously to AIC as

$$\begin{aligned}\text{DIC} &= \text{googness of fit} + \text{model complexity} \\ &= D(\bar{\boldsymbol{\theta}}) + 2(\boldsymbol{D} - D(\bar{\boldsymbol{\theta}})) = \bar{D} + p_D, \end{aligned} \tag{5.13}$$

where $p_D = \bar{D} - D(\bar{\boldsymbol{\theta}})$. If there are two or more competing models, then the model with the smallest DIC is estimated to be the preferred model in terms of short-term predictive ability. p_D can be negative when there is substantial conflict between prior and data, or when the posterior distribution for a parameter is extremely asymmetric or bimodal. Thus, a negative p_D is indicative of a poorly designed model (Spiegelhalter et al. 2002).

To check adequacy of the time-to-event model, we can use hazard-based residuals. From Equation (5.11), we see that the cumulative hazard for individual i is

$$H(t_i|\boldsymbol{\theta}) = \kappa\gamma\rho e^{\mathbf{z}_i'\beta} \int_0^{t_i} \frac{(1-e^{-s^\kappa})^{\gamma-1} e^{-s^\kappa} s^{\kappa-1}}{1-(1-e^{-s^\kappa})^{\gamma}} e^{\phi \mathbf{w}_i'(s)\mathbf{b}_i} ds \tag{5.14}$$

The posterior mean or median of (5.14) can be used as a point estimate of $H(t_i|\boldsymbol{\theta})$. We can then use the graphical technique described in Section 4.6.1 to check adequacy of the fitted model.

5.4.7 Software Implementation

As highlighted by Rizopoulos (2012), fitting of joint models is a highly demanding task. The computationally intensive Bayesian implementation of the methodology is described in this section. In implementing the Gibbs algorithm, we wrote our code in WinBUGS, which is a statistical software for Bayesian analysis using MCMC methods (Lunn et al. 2000). Since MCMC methods are computationally expensive, we used R (R Core Team 2018) for data manipulation and organization, and WinBUGS for MCMC implementation only. Manipulation in R includes organizing data for both longitudinal and time-to-event processes, generating initial values of the parameters to be used in WinBUGS, defining the nodes and weights of the Gauss-Legendre quadrature for numerical integration (see Equation (5.11)), defining the hyperprior values of the parameters, and

finally arranging and organizing all data we need to run MCMC in WinBUGS. The "R2WinBUGS" package (Sturtz et al. 2005) is then used to call WinBUGS from R. When the MCMC sampling is completed in WinBUGS, we again use R to summarize posterior characteristics of the parameters using the "coda" package (Plummer et al. 2006). Note that the two processes (longitudinal and event time) were first fitted separately using the maximum likelihood method, and the estimates from these fits were subsequently used as initial values for MCMC sampling.

There are also computational difficulties in WinBUGS implementation of our model because of floating point problems (Mäechler 2019). Such problems are caused by the internal representation of floating point numbers: all computing software use a fixed number of binary digits to represent a decimal number; some decimal numbers cannot be represented exactly in binary, resulting in small roundoff errors. We encountered such problems in the computation of the terms $(\gamma - 1)\log(1 - e^{-t_i^\kappa})$ and $\log\{1 - (1 - e^{-t_i^\kappa})^\gamma\}$ of our log-likelihood function (5.11). In particular, if $(1 - e^{-t_i^\kappa})^\gamma$ is very close to 1, WinBUGS makes $\log\{1 - (1 - e^{-t_i^\kappa})^\gamma\}$ an undefined number in floating point operations, rendering a computational breakdown of the MCMC. This problem is countered by using an approximation. We see that

$$(\gamma - 1)\log(1 - e^{-t_i^\kappa}) - \log\{1 - (1 - e^{-t_i^\kappa})^\gamma\} = -\log\left\{(1 - e^{-t_i^\kappa})^{1-\gamma} - (1 - e^{-t_i^\kappa})\right\}, \tag{5.15}$$

which reduces to t_i^κ if $(1-e^{-t_i^\kappa})^{1-\gamma} = 1$. Thus, we take Equation (5.15) equals to t_i^κ if $(1-e^{-t_i^\kappa})^{1-\gamma} \approx 1$ (exactly 1 in WinBUGS), otherwise we take $\min\{0.99999999999999, (1 - e^{-t_i^\kappa})^\gamma\}$ for $(1 - e^{-t_i^\kappa})^\gamma$ to calculate (5.15). Note that WinBUGS rounds any value between 0.99999999999999 and 1, and therefore we use the approximation $\min\{0.99999999999999, (1-e^{-t_i^\kappa})^\gamma\}$ for $(1-e^{-t_i^\kappa})^\gamma$. A numerical analysis reveals that the approximation is reasonably accurate. Our WinBUGS codes to fit joint models using the MKumW and Weibull distributions are presented in Appendix A.7.

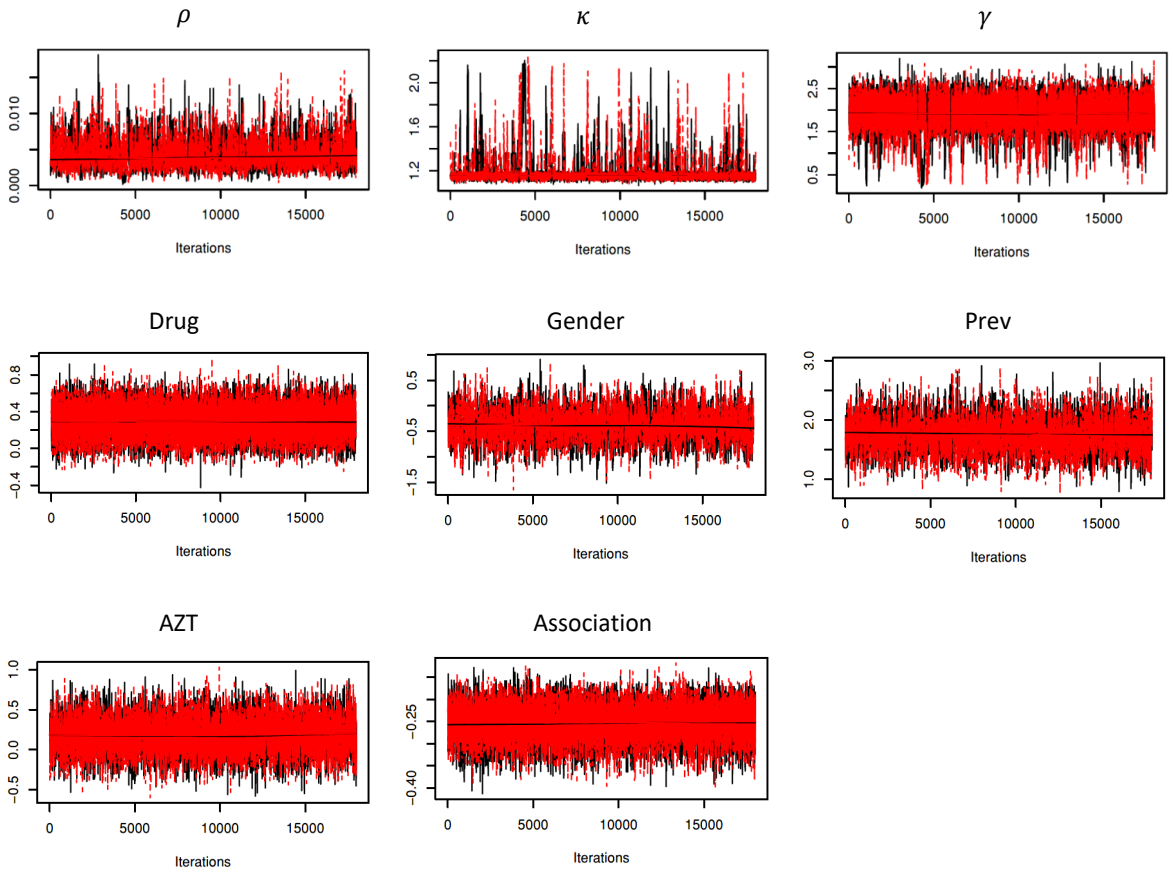

Figure 5.3: Trace plots of time-to-event sub-model (MKumW) parameters in Bayesian fit of the joint model to AIDS data.

5.4.8 An Application to AIDS Data

This example illustrates the application of the MKumW distribution in joint modeling of time-to-event and longitudinal data. Abrams et al. (1994) described a study involving 467 human immunodeficiency virus (HIV) infected patients who had failed or were intolerant to zidovudine therapy (AZT). The main objective was to compare two antiretroviral drugs to prevent the progression of HIV infections: didanosine (ddI) and zalcitabine (ddC). Patients were randomly assigned to receive either ddI or ddC and followed until death or the end of the study, resulted in 188 complete and 279 censored observations. It was also of interest to quantify the association between CD4 cell

counts (internal time-dependent covariate) and time to death. In addition, two more factors were thought possibly to be relevant to an individual's prognosis: gender and previous opportunistic infection (AIDS diagnosis).

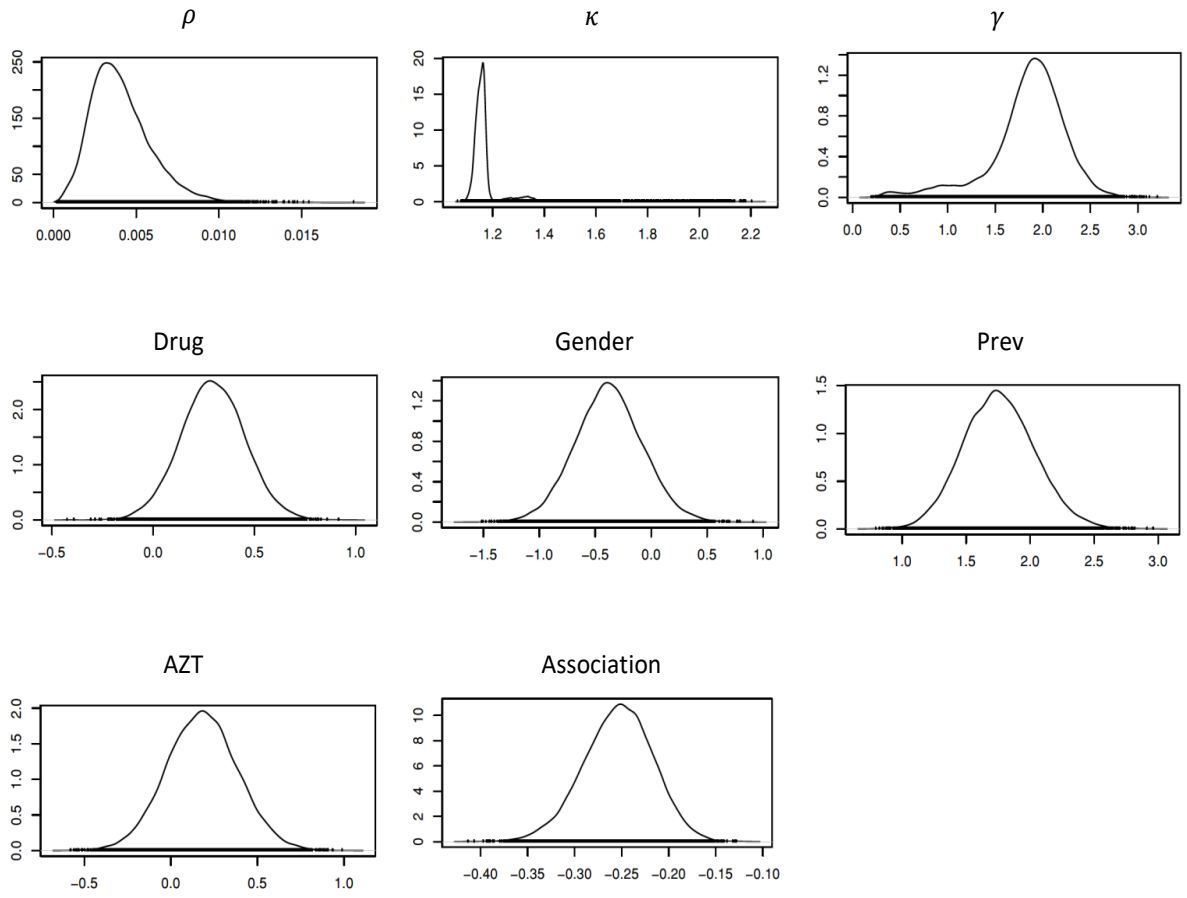

Figure 5.4: Density plots of time-to-event sub-model (MKumW) parameters in Bayesian fit of the joint model to AIDS data.

We let y_{ij} denote the square root of the j^{th} CD4 count measurement for patient i ($i = 1, 2, \ldots, n$ and $j = 1, 2, \ldots, n_i$). Four binary covariates are considered in our analysis: Drug = I(drug = ddI), Gender = I(gender = male), AZT = I(AZT failure), and Prev = I(previous opportunistic infection at study entry). The longitudinal response y_{ij} at time s_{ij} is modeled using the linear mixed-effects model

$$y_{ij} = \mathbf{x}'_{ij}\boldsymbol{\alpha} + \mathbf{w}'_{ij}\boldsymbol{b}_i + \epsilon_{ij}, \tag{5.16}$$

where

$$\mathbf{x}'_{ij}\boldsymbol{\alpha} = \alpha_0 + \alpha_1 s_{ij} + \alpha_2(s_{ij} \times \text{Drug}_i) + \alpha_3 \text{Gender}_i + \alpha_4 \text{Prev}_i + \alpha_5 \text{AZT}_i, \tag{5.17}$$

$$\mathbf{w}'_{ij}\mathbf{b}_i = b_{i0} + b_{i1} s_{ij}. \tag{5.18}$$

The time-to-event model is specified as

$$h(t_i) = h_0(t_i; \boldsymbol{\zeta}) \exp\{\mathbf{z}'_i\boldsymbol{\beta} + V_i(t)\}, \tag{5.19}$$

where

$$\mathbf{z}'_i\boldsymbol{\beta} = \beta_1 \text{Drug}_i + \beta_2 \text{Gender}_i + \beta_3 \text{Prev}_i + \beta_4 \text{AZT}_i, \tag{5.20}$$

$$V_i(t) = \phi(b_{i0} + b_{i1} t). \tag{5.21}$$

For MCMC analysis, we construct two Markov chains each of 100,000 iterations to approximate the posterior density. The initial 10,000 iterations are discarded as burn-in, and the inferences are based on every 5^{th} iteration of the chains (thinning), resulting in a total of 18,000 iterations per chain. Trace and density plots of the time-to-event sub-model parameters are displayed in Figures 5.3 and 5.4, respectively, whereas those of the longitudinal sub-model parameters are shown in Figures 5.5 and 5.6, respectively. The lack of any trend in the trace plots in the two chains indicate good mixing, and the density plots display no signs of multimodality. The Gelman-Rubin statistic also suggests good mixing and convergence of the chains, with values less than 1.01 for all the parameters and quantities of interest. We also considered the Weibull distribution for joint modeling; the objective is to compare the performance of the two models in terms of goodness-of-fit. Note that Weibull is the most widely used time-to-event model for joint analysis. However, our future plan is to implement the piecewise linear and Cox PH model in WinBUGS for further comparison.

DIC values of the MKumW and Weibull fits are 7614.9 and 7628.83, respectively, suggesting

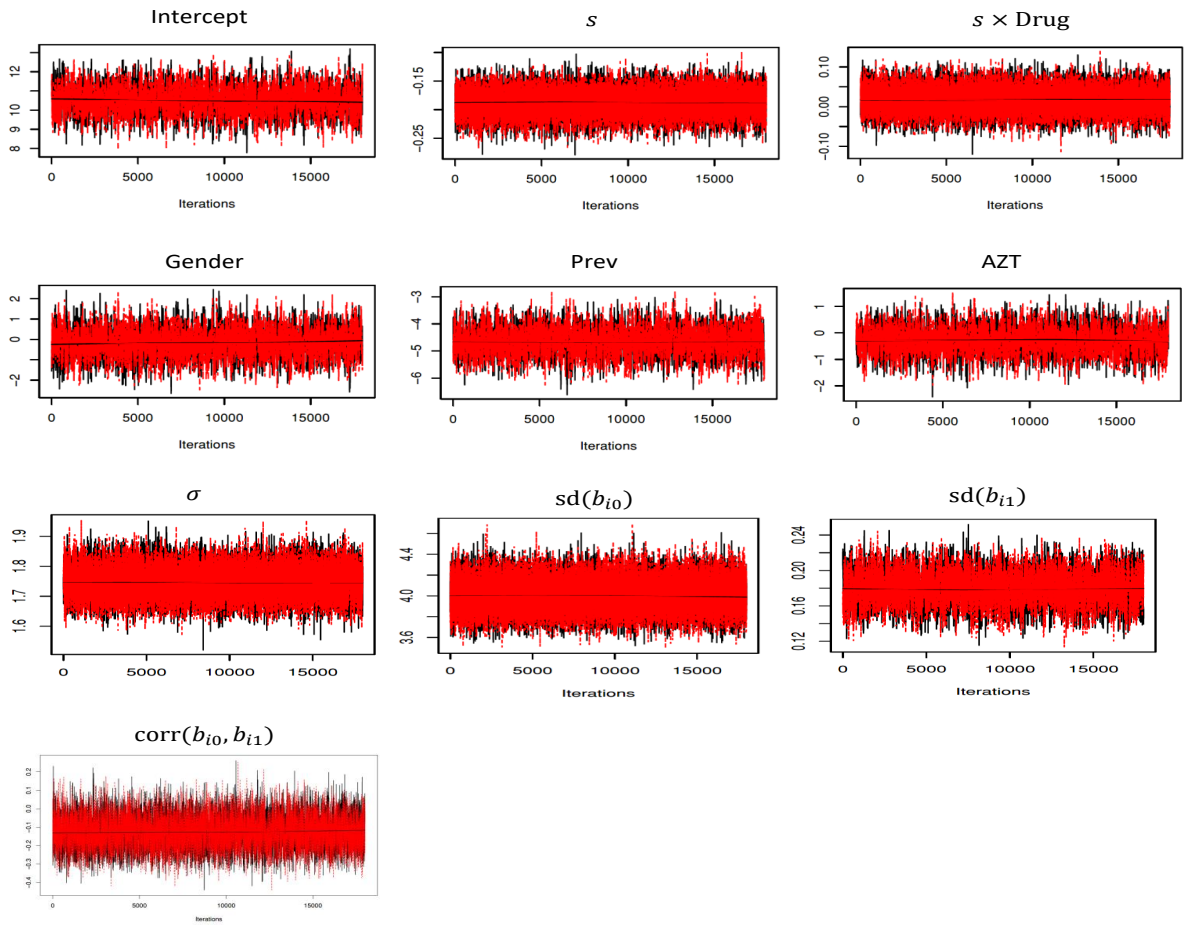

Figure 5.5: Trace plots of longitudinal sub-model (linear mixed-effects) parameters in Bayesian fit of the joint model to AIDS data.

that the MKumW fit is superior to that of the Weibull for the AIDS data. The residual plot (Figure 5.7) leads to the same conclusion: residuals lying closely to the unit-slope line for the MKumW indicate its superiority over the Weibull model. Note that residual analysis has not been extensively studied for joint models. Although we consider residual analysis for the time-to-event process, we have not explored checking model adequacy of the longitudinal process in this thesis. It is one of our future research goals.

Some posterior characteristics of parameters for the two models (MKumW and Weibull) are given in Table 5.3. Although the estimates of the regression coefficients are comparable for the two models, the flexibility provided by an additional shape parameter of the MKumW distribution

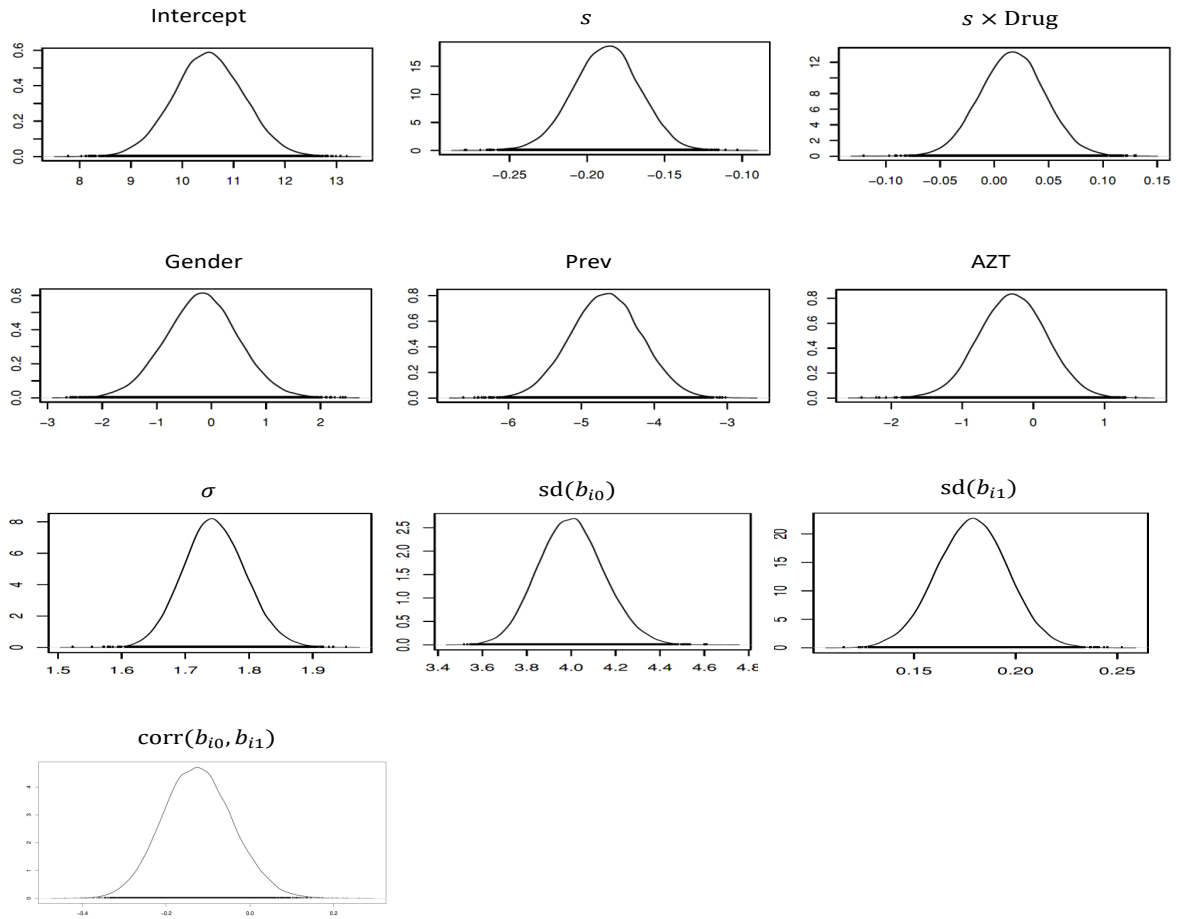

Figure 5.6: Density plots of longitudinal sub-model (linear mixed-effects) parameters in Bayesian fit of the joint model to AIDS data.

leads to its overall superiority over the Weibull model. For the time-to-event process, 95% credible intervals indicate significant effects for previous opportunistic infection (Prev) and CD4 cell counts (95% credible intervals for Prev and CD4 cell counts are $(1.230, 2.325)$ and $(-0.330, -0.185)$, respectively, both of which exclude 0). In particular, the estimate of the regression coefficient for CD4 is -0.253, suggesting that 1 unit increase in the square root of CD4 cell counts reduce the relative risk of death about 22% ($\exp(-0.253) = 0.776$), controlling for the other factors. Similarly, an individual with a history of previous opportunistic infection is 5.76 times ($\exp(1.751) = 5.76$) more likely to die than an individual with no history of previous opportunistic infection, controlling for the other factors. The longitudinal process also suggests that Prev is highly significant (negative

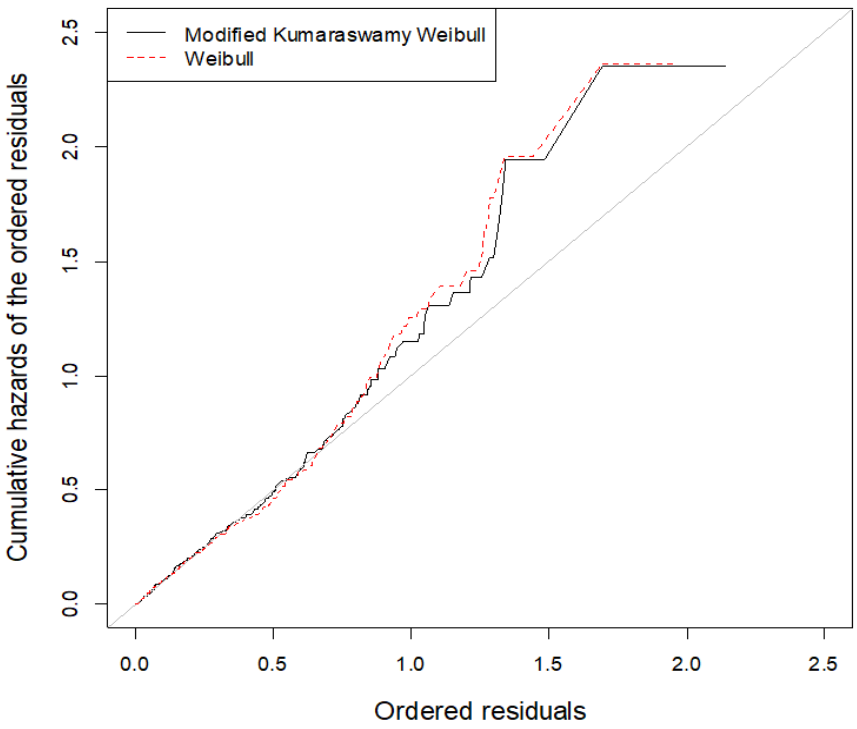

Figure 5.7: Residual plots for MKumW and Weibull joint models for AIDS data (black for MKumW and red for Weibull).

effect) for CD4 cell counts. The posterior characteristics of the standard deviations suggest small variations in b_{i0} and b_{i1} (SD(b_{i0}) = 3.998 and SD(b_{i1}) = 0.179), that is, (a) a small variation in patients at study entry, and (b) a small variation in patients with respect to progression of the disease over time. However, there is no significant association between study entry characteristics and the progression of the disease (with respect to CD4 cell counts) over time (Corr(b_{i0}, b_{i1}) = −0.126 with 95% credible interval $(-0.283, 0.044)$).

Table 5.3: Posterior summaries of the joint model parameters in Bayesian fits to the AIDS data.

		MKumW PH (DIC = 7614.9)			Weibull PH (DIC = 7628.83)		
	Parameter	Posterior Median	Standard Deviation	95% Credible Interval	Posterior Median	Standard Deviation	95% Credible Interval
Time-to-event process	β_1 (Drug)	0.294	0.157	$-0.015, 0.607$	0.293	0.158	$-0.017, 0.601$
	β_2 (Gender)	-0.381	0.301	$-0.969, 0.220$	-0.377	0.293	$-0.939, 0.215$
	β_3 (Prev)	1.751	0.279	$1.230, 2.325$	1.707	0.276	$1.193, 2.276$
	β_3 (AZT)	0.180	0.205	$-0.216, 0.587$	0.176	0.201	$-0.212, 0.575$
	ϕ (CD4)	-0.253	0.037	$-0.330, -0.185$	-0.245	0.036	$-0.319, -0.179$
	ρ	0.004	0.002	$0.001, 0.008$	0.002	0.001	$0.001, 0.005$
	κ	1.158	0.143	$1.116, 1.655$	1.492	0.099	$1.305, 1.694$
	γ	1.902	0.399	$0.774, 2.484$			
Longitudinal process	α_0 (Intercept)	10.510	0.681	$9.180, 11.850$	10.510	0.667	$9.189, 11.800$
	α_1 (s)	-0.187	0.021	$-0.230, -0.147$	-0.186	0.021	$-0.229, -0.144$
	α_2 (s×Drug)	0.017	0.030	$-0.041, 0.075$	0.017	0.030	$-0.041, 0.077$
	α_3 (Gender)	-0.171	0.663	$-1.478, 1.137$	-0.164	0.654	$-1.427, 1.125$
	α_4 (Prev)	-4.660	0.480	$-5.597, -3.723$	-4.670	0.474	$-5.604, -3.754$
	α_5 (AZT)	-0.290	0.472	$-1.220, 0.628$	-0.301	0.466	$-1.192, 0.626$
	σ	1.745	0.049	$1.653, 1.844$	1.744	0.049	$1.654, 1.844$
	SD(b_{i0})	3.998	0.147	$3.725, 4.299$	3.996	0.147	$3.726, 4.297$
	SD(b_{i1})	0.179	0.017	$0.144, 0.213$	0.179	0.018	$0.144, 0.213$
	Corr(b_{i0}, b_{i1})	-0.126	0.084	$-0.283, 0.044$	-0.126	0.084	$-0.284, 0.047$

5.5 Conclusion

In Chapter 4, we proposed the three-parameter MKumW distribution, which is closed under the PH relationship. We then formulated a PH regression model based on the proposed distribution, and developed large sample theory for statistical inference. Simulation study and real data example revealed that the proposed model could be valuable in adequately describing different types of time-to-event data.

Applications of PH models are also commonly seen in recurrent event data analysis (Cook and Lawless 2007) and joint modeling of longitudinal and time-to-event data (Rizopoulos 2012). In particular, a flexible parametric model is desirable in joint modeling of longitudinal and time-to-event processes. The rationale behind this is that a full likelihood approach must be employed for joint modeling. Although the Cox model has many appealing features, the maximum likelihood

method for the Cox model is based on a partial likelihood function, as the model is semi-parametric with the baseline hazard function assumed an arbitrary non-negative function of time. Therefore, the Cox model cannot be applied directly in joint modeling theory and applications. This problem is handled by replacing the unspecified cumulative hazard function by a step function with jumps at the unique event times. The difficulty with this technique is that it involves a very high-dimensional parameter vector, which often leads to numerical complications in estimation as well as inefficient estimates of the parameters (the standard errors are generally underestimated). Although bootstrapping may be used to estimate standard errors, it is also computationally demanding (for a detail discussion about this topic, readers may refer to Rizopoulos (2012)). Based on this ground, the proposed MKumW model may play a very important role in joint modeling theory and applications.

In this chapter, we initially developed large sample likelihood-based methods for recurrent event data modeling using the MKumW distribution. Then, we proposed a Bayesian approach for joint modeling of longitudinal and time-to-event data. We also developed algorithms for computationally intensive Bayesian approach, implemented in WinBUGS and R. Overall, this work could be valuable in recurrent event and joint modeling theory and applications.

Chapter 6

CONCLUDING REMARKS

There are many approaches to formulate regression models for time-to-event data, including accelerated failure time models where the covariates effectively alter the time scale, and proportional hazards (PH) model where the covariates affect the hazard function. In particular, PH models are extremely popular in clinical trials and medical studies, perhaps because of the relative risk interpretation of the regression coefficients, and the existence of a semi-parametric proportional hazards model (i.e., the Cox PH model) which is robust against the distributional assumption of the survival time. The focus of this thesis is on the PH family, with particular emphasis on parametric models.

In general, a parametric model is preferred in statistical data analysis if the underlying distributional assumption is reasonably accurate. Parametric models in time-to-event data analysis can lead to more efficient estimates of the regression coefficients than the semi-parametric Cox PH model, as demonstrated by Efron (1977) and Oakes (1977). Parametric models also allow estimates of other relevant quantities efficiently, including quantiles, survival probabilities and the hazard function. Note that the hazard function represents an important aspect of the time course of a disease process, and therefore is often of fundamental interest in many clinical studies (Royston and Parmar 2002). In addition, parametric models are pivotal in joint modeling of longitudinal and time-to-event data, because the use of the Cox PH in joint modeling usually leads to an underestimation of the standard errors of the parameter estimates (Hsieh et al. 2006, Rizopoulos 2012). However, the parametric options for PH models are very limited. Among these, the Weibull and piecewise constant hazards are widely used in the literature; the Weibull distribution is par-

simonious but can characterize only monotone hazard functions, whereas the piecewise constant hazards model may be flexible but may involve many parameters to estimate, depending on the number of partitions of the time scale that makes the hazard function piecewise constant. Thus, with the increasing availability of data with wide ranging characteristics, it is desirable to formulate flexible and parsimonious PH models. In this thesis, we propose two parametric PH models for time-to-event data, and develop theory for statistical inference. As demonstrated, the proposed models are fairly flexible and parsimonious, and can be valuable in survival analysis theory and applications. Perhaps the most important contribution of this thesis involves further extension of one of the proposed models to recurrent event data analysis and joint modeling of longitudinal and time-to-event data.

In Section 6.1, we present a summary of the methods proposed in this thesis. Some cautionary remarks and future research directions are then presented in Sections 6.2 and 6.3, respectively.

6.1 Contribution of the Thesis: A Summary

In this thesis, we proposed two fully parametric models with sufficient flexibility to incorporate different types of hazard shapes: the generalized log-logistic distribution, and the modified Kumaraswamy Weibull (MKumW) distribution. We then tailored the MKumW distribution to recurrent event data analysis using the maximum likelihood method and joint modeling of longitudinal and time-to-event data using a Bayesian approach.

6.1.1 The Generalized Log-Logistic Distribution

The log-logistic distribution has wide applications in analyzing survival data. In particular, it is useful to describe data which exhibit unimodal hazard shape (Lawless 2003). The model is closed under both multiplication of failure time and proportionality of odds. However, it is not a proportional hazard (PH) model. In Chapter 3, we propose a generalization of the log-logistic

distribution. The proposed model is a three-parameter distribution, and has characteristics similar to those of the log-logistic model. Moreover, it approaches the Weibull in the limit. These features enable it to satisfactorily handle both monotone (increasing and decreasing) and nonmonotone (unimodal) hazard functions. The performance of the proposed model has been demonstrated with application to four data sets, one of which involves joint modeling of time-to-event and longitudinal data. It turns out that the generalized log-logistic may provide better fits in describing unimodal hazard functions compared to the log-logistic distribution. A comparison between the generalized log-logistic, Weibull and the Cox PH models via simulations reveals that the generalized log-logistic PH model performs reasonably well in analyzing different types of time-to-event data. The flexibility provided by the generalized log-logistic model could be very useful in adequately describing different types of time-to-event data. An article on this topic has been published in the *Journal of Statistical Distributions and Applications* (Khan and Khosa 2016).

6.1.2 The Modified Kumaraswamy Weibull (MKumW) distribution

Aside from the fact that the generalized log-logistic model does not naturally accommodate monotone increasing hazard shapes, it is also not flexible enough to deal with bathtub-shaped hazard functions. Note that the bathtub-shaped hazard function is widely used to describe the process of human life and data in reliability engineering (e.g., failure rate of an electronic component). With this motivation, we propose a more general parametric PH model (Chapter 4), which is parsimonious and flexible in the sense that it accommodates all four standard shapes of the hazard function (increasing, decreasing, unimodal and bathtub shape) at the small cost of estimating only three distributional parameters. The model is derived as a special case of the Kumaraswamy Weibull distribution (Cordeiro et al. 2010), and accommodates the Weibull model as a special case. We then formulate a regression model based on the proposed distribution, and develop large sample theory for statistical inference. A simulation study and a real data example reveal that the proposed model can be very useful in analyzing survival data with ranging characteristics. Simulation

studies are important to evaluate the performance of new methodologies. The general performance of the proposed models under various assumptions has been evaluated through a simulation study in which we generate realistic datasets. More specifically, we generate data under increasing, decreasing, unimodal and bathtub shape hazard functions, and analyzed each of these using several competing models. This allow us is to discuss the performance of the our proposed methodology under realistic situations.

6.1.3 Recurrent Event Data Analysis and Joint Modeling

Recurrent event data analysis and joint modeling of longitudinal and time-to-event data are two very important areas of research in survival analysis, as many data from different applications fall under these setups. Recurrent event data arise when the individuals under study may experience multiple events over time, with the primary objective to explore the effects of fixed and/or time-dependent covariates to the occurrences of the events of interest. On the other hand, the joint modeling problem involves describing a process where the event time distribution depends on a longitudinally measured internal covariate (e.g., prostate specific antigen (PSA) measurements may be obtained for patients following treatment for prostate cancer (longitudinally measured internal covariate), along with time to disease recurrence). We considered theoretical development of these two problems based on our flexible MKumW model in Chapter 5.

For recurrent event data analysis, the novelty of our methodology lies in formulating a recurrent event model based on the MKumW distribution, and developing theory for statistical inference using the maximum likelihood method. For joint modeling, the novelty lies in formulating a hierarchical model based on the MKumW distribution, proposing a Bayesian approach for statistical inference, and computationally intensive Bayesian implementation of the methodology in the statistical software WinBUGS. Note that a key computational difficulty in fitting joint models is that the model involves multiple integrals that do not have analytical solutions. This problem was dealt with approximating the integrals using a numerical approach; the implementation of this numerical

method in WinBUGS is another novelty of our work.

Comparative studies based on real data analyses indicate better performance of the proposed methods in comparison with the Weibull, the most commonly used parametric PH model in survival analysis. Overall, this work could be valuable in recurrent event and joint modeling theory and applications.

6.2 Cautionary Remarks

Because of the flexibility of our methodology, it could serve as a powerful statistical tool in analyzing time to event data. However, some caution is required for the following reasons.

- The performance of a parametric model depends on the validity of the underlying distributional assumption. Note that parametric models may lead to misleading conclusions and statistical inferences if the distributional assumption is not satisfied. For example, the generalized log-logistic model should not be used when the data exhibit bathtub-shape failure rates, as it is theoretically not capable of describing such data. Therefore, it is important to check the appropriateness of the chosen distribution in describing the data under study. Overall, a fully parametric model involves stronger assumptions than, for example, the semi-parametric Cox model. Thus, a richer parametric model or simply the Cox model should be used in case of an unsatisfactory fit of the chosen probability distribution.

- Fitting time-to-event models based on the MKumW distribution is computationally difficult because of the floating-point problems (see Section 5.4.7). Such problems are caused due to round off errors in computing software. For likelihood-based inference, we used the Rmpfr package (Maechler 2016) in R to generate high precision numbers to deal with this problem. Although use of high precision numbers is important for computational accuracy, it is not efficient in terms of computational speed. Note that there is no way to obtain high precision

numbers in the WinBUGS software for Bayesian inference. For this reason, we used some approximations to fit joint models in WinBUGS as described in Section 5.4.7

- The MCMC method for joint models is computationally expensive, and may need a considerable amount of time for such a process to complete. To fit the proposed joint model to the AIDS data with 467 subjects, WinBUGS took about 339 minutes using Intel i7-4770 CPU with 32 GB RAM. We are considering a parallel implementation of the WinBUGS model for faster Bayesian inference (see Section 6.3).

  .

6.3 Future work

There are several considerations to extend the work presented in this thesis. Some potential future research topics are described as follows.

6.3.1 Theoretical Work

1. The recurrent event model of Chapter 4 was developed based on the Poisson process formulation (i.e., event counts). In such a process, the occurrences of events are assumed to have no effects on the process itself (e.g., mild asthmatic attacks in humans; see Cook and Lawless (2007) for more details). In many studies, it is not reasonable to assume that the events are incidental. For example, the occurrence of a stoke may alter the process over time (i.e., more likely to recur the event compared to a healthy subject). For such a process, some type of individual renewal occurs after an event, and the events are relatively infrequent. A renewal process is recommended to describe such processes (Cook and Lawless 2007), where analyses are based on gap times between successive occurrences of the events as opposed to simply event counts. Developing recurrent event methodology for the MKumW distribution based on gap times could be an area of future research.

2. Joint modeling continues to be an active research area, particularly due to various details that are not part of standard data analysis, and due to computational challenges. Some theoretical work based on the proposed distributions in jointly of modeling longitudinal and time-to-event are discussed below.

    - In time-to-event data analyses, estimates of the survival probabilities and quantiles (for example, median survival time) could be of particular interest to the researchers. Estimation of these quantities is relatively simpler in analyzing standard survival data using parametric models. However, estimation of such quantities for joint models is highly complicated, and could be an area of future research. The idea is to derive expressions for survival function and quantiles for the proposed model based on the MKumW distribution, and then use the MCMC samples to summarize their posterior distributions. Note that the survival function $S(t)$ for joint models involves integrals with respect to time, and the estimation of the, for example, median requires to solve the equation $S(t) = 0.5$. Thus, estimation of these quantities involves both theoretical and computational challenges.

    - In this study, we emphasized on the joint inference for a single endogenous time-dependent covariate and the time-to-event data with single outcome/failure. However, in many clinical and epidemiological studies more than one cause of failure is possible. For example, one may be interested in time to death due to breast cancer, some patient can die from breast cancer or from stroke, but he cannot die from both. In such situations death from more than one cause of failure can be considered as competing risks. Tailoring proposed models to joint modeling set-up to study the association between longitudinal and competing risks survival data could be another promising area of future research.

    - In many studies, interest lies in quantifying the association between longitudinal binary

outcomes (internal covariates) and time-to-event data. When data from two or more longitudinal processes are available, another scientific focus is on the degree to which changes in one process are associated with changes in another process. Developing joint models using the proposed distributions to investigate the effects of correlated longitudinal binary responses on time-to-event data could be an area of future research topic.

- Many longitudinal studies involve trajectories that exhibit nonlinearity over time. It could be of particular interest to develop joint models based on the proposed distributions where the longitudinal profiles exhibit nonlinearity. This consideration might be another area of future research.

- In the standard linear mixed-effects models, subject-specific residuals and the marginal residuals are often used to assess the assumptions of the longitudinal part of the joint model. Thus, to check the validity of the assumptions of the longitudinal part of proposed joint models will be considered for future research.

6.3.2 Computational Work

1. We initially considered likelihood approach to fit joint models (Chapter 3). Specifically, we modified the source codes of the JM package (Rizopoulos et al. 2010) to fit joint models using the generalized log-logistic baseline hazard function. With this technique, the optimization is somewhat sensitive to the choice of the initial values of the parameters (i.e., sometimes fails to locate a maximum with the given set of initial values). In particular, fitting joint models using the MKumW distribution is quite sensitive to the choice of the initial values. This is one of the reasons to consider a Bayesian approach to fit joint models with the MKumW distribution, as described in Chapter 4. In the same spirit, we plan to develop a Bayesian approach for joint models using the generalized log-logistic distribution.

2. A new Bayesian software MultiBUGS has been developed recently, which allows parallel computation. As such, it substantially speeds up posterior inference of Bayesian models. However, an R interface for MultiBUGS is under development. We plan to implement our approach for joint modeling in MultiBUGS.

3. An R package to fit joint models is in progress. This package will allow other researchers to implement the methods for their own datasets relevant to this thesis' objectives. Such implementation is important, as otherwise any publication of the work out of this thesis may never be used outside of the statistical community.

6.3.3 Publication

As mentioned above, one article based on the generalized log-logistic PH model has been published in *Journal of Statistical Distributions and Applications* (Khan and Khosa 2016). Another article based on the MKumW model and its application in recurrent event data analysis and joint modeling of time-to-event and longitudinal data is in progress. A third article would be on the software implementation of the proposed models, possibly in a journal in computational statistics.

Appendix

A.1 Generalized Log-Logistic Model: Derivatives of the Log-Likelihood Function

Let $m = \sum_{i=1}^{n} \delta_i$, $a_i = \exp(\mathbf{z}_i'\boldsymbol{\beta})$, $b_i = (\gamma t_i)^\kappa$, $c_i = \log b_i/(1+b_i)$ and $d_i = b_i/(1+b_i)$. We have

- $$\log(\gamma t_i) = \frac{\log b_i}{\kappa}, \tag{A.1}$$
- $$(\gamma t_i)^\kappa \log(\gamma t_i) = \frac{b_i \log b_i}{\kappa}, \tag{A.2}$$
- $$\frac{\partial b_i}{\partial \kappa} = \frac{\partial}{\partial \kappa}(\gamma t_i)^\kappa = (\gamma t_i)^\kappa \log(\gamma t_i) = \frac{b_i \log b_i}{\kappa}, \tag{A.3}$$
- $$\frac{\partial \log b_i}{\partial \kappa} = \frac{\log b_i}{\kappa}, \tag{A.4}$$
- $$\frac{\partial \log(1+b_i)}{\partial \kappa} = \frac{b_i \log b_i}{\kappa(1+b_i)} = \frac{b_i c_i}{\kappa}, \tag{A.5}$$
- $$\frac{\partial b_i \log b_i}{\partial \kappa} = \frac{b_i \log b_i}{\kappa} \log b_i + b_i \frac{\log b_i}{\kappa} = \frac{b_i (\log b_i)(1+\log b_i)}{\kappa}, \tag{A.6}$$
- $$\frac{\partial c_i}{\partial \kappa} = \frac{\partial}{\partial \kappa} \frac{\log b_i}{1+b_i} = \frac{\log b_i}{\kappa(1+b_i)} \Big(1 - \frac{b_i \log b_i}{1+b_i}\Big) = \frac{c_i(1-b_i c_i)}{\kappa}, \tag{A.7}$$
- $$\frac{\partial b_i c_i}{\partial \kappa} = \frac{b_i c_i (1 - b_i c_i + \log b_i)}{\kappa} = \frac{b_i c_i (1+c_i)}{\kappa}, \tag{A.8}$$
- $$\frac{\partial d_i}{\partial \kappa} = \frac{\partial}{\partial \kappa} \frac{b_i}{1+b_i} = \frac{b_i \log b_i}{\kappa(1+b_i)} \Big(1 - \frac{b_i}{1+b_i}\Big) = \frac{c_i d_i}{\kappa}, \tag{A.9}$$
- $$\frac{\partial \log(1-d_i)}{\partial \kappa} = \frac{\partial}{\partial \kappa} \log(1+b_i)^{-1} = -\frac{\partial}{\partial \kappa} \log(1+b_i) = -\frac{b_i c_i}{\kappa}, \tag{A.10}$$
- $$\frac{\partial b_i}{\partial \gamma} = \frac{\partial}{\partial \gamma}(\gamma t_i)^\kappa = \kappa \gamma^{\kappa-1} t_i^\kappa = \frac{\kappa}{\gamma} b_i, \tag{A.11}$$

- $$\frac{\partial d_i}{\partial \gamma} = \frac{\partial}{\partial \gamma}\frac{b_i}{1+b_i} = \frac{\kappa}{\gamma}\frac{b_i}{1+b_i}\Big(1 - \frac{b_i}{1+b_i}\Big) = \frac{\kappa}{\gamma}\, d_i(1-d_i), \tag{A.12}$$

- $$\frac{\partial}{\partial \gamma}\log(1-d_i) = -\frac{\partial}{\partial \gamma}\log(1+b_i) = -\frac{\kappa}{\gamma}\,\frac{b_i}{1+b_i} = -\frac{\kappa}{\gamma}\, d_i. \tag{A.13}$$

Using (3.6) and (A.1)-(A.13), we can derive the first and second derivatives of the log-likelihood function as follows.

$$\begin{aligned}\frac{\partial \ell(\boldsymbol{\theta})}{\partial \kappa} &= \frac{m}{\kappa} + m\log\rho + \sum_{i=1}^{n}\delta_i \log t_i - \frac{1}{\kappa}\sum_{i=1}^{n}\delta_i b_i c_i - \Big(\frac{\rho}{\gamma}\Big)^{\kappa}\Big(\frac{1}{\kappa}\Big)\sum_{i=1}^{n} a_i b_i c_i \\ &\quad - \Big(\frac{\rho}{\gamma}\Big)^{\kappa}\log\Big(\frac{\rho}{\gamma}\Big)\sum_{i=1}^{n} a_i \log(1+b_i).\end{aligned}$$

$$\frac{\partial \ell(\boldsymbol{\theta})}{\partial \gamma} = -\Big(\frac{\kappa}{\gamma}\Big)\sum_{i=1}^{n}\delta_i d_i - \Big(\frac{\kappa}{\gamma}\Big)\Big(\frac{\rho}{\gamma}\Big)^{\kappa}\sum_{i=1}^{n} a_i d_i - \Big(\frac{\kappa}{\gamma}\Big)\Big(\frac{\rho}{\gamma}\Big)^{\kappa}\sum_{i=1}^{n} a_i \log(1-d_i).$$

$$\frac{\partial \ell(\boldsymbol{\theta})}{\partial \rho} = \frac{m\kappa}{\rho} - \Big(\frac{\kappa}{\rho}\Big)\Big(\frac{\rho}{\gamma}\Big)^{\kappa}\sum_{i=1}^{n} a_i \log(1+b_i).$$

$$\frac{\partial \ell(\boldsymbol{\theta})}{\partial \beta_j} = \sum_{i=1}^{n}\delta_i z_{ij} - \Big(\frac{\rho}{\gamma}\Big)^{\kappa}\sum_{i=1}^{n} a_i \log(1+b_i) z_{ij} \quad \text{for } j = 1, 2, \ldots, p.$$

$$\begin{aligned}\frac{\partial^2 \ell(\boldsymbol{\theta})}{\partial \kappa^2} &= \frac{\partial}{\partial \kappa}\frac{\partial \ell(\boldsymbol{\theta})}{\partial \kappa} \\ &= -\frac{m}{\kappa^2} + \frac{1}{\kappa^2}\sum_{i=1}^{n}\delta_i b_i c_i - \frac{1}{\kappa^2}\sum_{i=1}^{n}\delta_i b_i c_i(1+c_i) - \Big(\frac{\rho}{\gamma}\Big)^{\kappa}\log\Big(\frac{\rho}{\gamma}\Big)\Big(\frac{1}{\kappa}\Big)\sum_{i=1}^{n} a_i b_i c_i \\ &\quad + \Big(\frac{\rho}{\gamma}\Big)^{\kappa}\Big(\frac{1}{\kappa^2}\Big)\sum_{i=1}^{n} a_i b_i c_i - \Big(\frac{\rho}{\gamma}\Big)^{\kappa}\Big(\frac{1}{\kappa^2}\Big)\sum_{i=1}^{n} a_i b_i c_i(1+c_i) \\ &\quad - \Big(\frac{\rho}{\gamma}\Big)^{\kappa}\log\Big(\frac{\rho}{\gamma}\Big)\Big(\frac{1}{\kappa}\Big)\sum_{i=1}^{n} a_i b_i c_i - \Big(\frac{\rho}{\gamma}\Big)^{\kappa}\Big\{\log\Big(\frac{\rho}{\gamma}\Big)\Big\}^2\sum_{i=1}^{n} a_i \log(1+b_i) \\ &= -\frac{m}{\kappa^2} - \frac{1}{\kappa^2}\sum_{i=1}^{n}\delta_i b_i c_i^2 - \Big(\frac{\rho}{\gamma}\Big)^{\kappa}\Big(\frac{1}{\kappa^2}\Big)\sum_{i=1}^{n} a_i b_i c_i^2 - \Big(\frac{2}{\kappa}\Big)\Big(\frac{\rho}{\gamma}\Big)^{\kappa}\log\Big(\frac{\rho}{\gamma}\Big)\sum_{i=1}^{n} a_i b_i c_i \\ &\quad - \Big(\frac{\rho}{\gamma}\Big)^{\kappa}\Big\{\log\Big(\frac{\rho}{\gamma}\Big)\Big\}^2\sum_{i=1}^{n} a_i \log(1+b_i).\end{aligned}$$

$$
\begin{aligned}
\frac{\partial^2 \ell(\boldsymbol{\theta})}{\partial \gamma^2} &= \frac{\partial}{\partial \gamma}\frac{\partial \ell(\boldsymbol{\theta})}{\partial \gamma} \\
&= \Big(\frac{\kappa}{\gamma^2}\Big)\sum_{i=1}^{n}\delta_i d_i - \Big(\frac{\kappa}{\gamma}\Big)^2\sum_{i=1}^{n}\delta_i d_i(1-d_i) + \kappa\rho^{\kappa}\Big(\frac{\kappa+1}{\gamma^{\kappa+2}}\Big)\sum_{i=1}^{n}a_i d_i \\
&\quad - \Big(\frac{\kappa}{\gamma}\Big)^2\Big(\frac{\rho}{\gamma}\Big)^{\kappa}\sum_{i=1}^{n}a_i d_i(1-d_i) + \kappa\rho^{\kappa}\Big(\frac{\kappa+1}{\gamma^{\kappa+2}}\Big)\sum_{i=1}^{n}a_i\log(1-d_i) \\
&\quad + \Big(\frac{\kappa}{\gamma}\Big)^2\Big(\frac{\rho}{\gamma}\Big)^{\kappa}\sum_{i=1}^{n}a_i d_i \\
&= \Big(\frac{\kappa}{\gamma^2}\Big)\sum_{i=1}^{n}\delta_i d_i - \Big(\frac{\kappa}{\gamma}\Big)^2\sum_{i=1}^{n}\delta_i d_i(1-d_i) \\
&\quad + \frac{\kappa(\kappa+1)}{\gamma^2}\Big(\frac{\rho}{\gamma}\Big)^{\kappa}\sum_{i=1}^{n}a_i[d_i+\log(1-d_i)] + \Big(\frac{\kappa}{\gamma}\Big)^2\Big(\frac{\rho}{\gamma}\Big)^{\kappa}\sum_{i=1}^{n}a_i d_i^2.
\end{aligned}
$$

$$
\frac{\partial^2 \ell(\boldsymbol{\theta})}{\partial \rho^2} = \frac{\partial}{\partial \rho}\frac{\partial \ell(\boldsymbol{\theta})}{\partial \rho} = -\frac{m\kappa}{\rho^2} - \frac{\kappa(\kappa-1)}{\rho^2}\Big(\frac{\rho}{\gamma}\Big)^{\kappa}\sum_{i=1}^{n}a_i\log(1+b_i).
$$

$$
\frac{\partial^2 \ell(\boldsymbol{\theta})}{\partial \beta_j \partial \beta_{j'}} = \frac{\partial}{\partial \beta_j}\frac{\partial \ell(\boldsymbol{\theta})}{\partial \beta_{j'}} = -\Big(\frac{\rho}{\gamma}\Big)^{\kappa}\sum_{i=1}^{n}a_i\log(1+b_i)z_{ij}z_{ij'} \quad \text{for } j, j' = 1, 2, \ldots, p.
$$

$$
\begin{aligned}
\frac{\partial^2 \ell(\boldsymbol{\theta})}{\partial \kappa \partial \gamma} &= \frac{\partial}{\partial \kappa}\frac{\partial \ell(\boldsymbol{\theta})}{\partial \gamma} \\
&= -\Big(\frac{1}{\gamma}\Big)\sum_{i=1}^{n}\delta_i d_i - \Big(\frac{\kappa}{\gamma}\Big)\Big(\frac{1}{\kappa}\Big)\sum_{i=1}^{n}\delta_i c_i d_i - \Big(\frac{\kappa}{\gamma}\Big)\Big(\frac{\rho}{\gamma}\Big)^{\kappa}\Big(\frac{1}{\kappa}\Big)\sum_{i=1}^{n}a_i c_i d_i \\
&\quad - \Big(\frac{1}{\gamma}\Big)\Big(\frac{\rho}{\gamma}\Big)^{\kappa}\sum_{i=1}^{n}a_i d_i - \Big(\frac{\kappa}{\gamma}\Big)\Big(\frac{\rho}{\gamma}\Big)^{\kappa}\log\Big(\frac{\rho}{\gamma}\Big)\sum_{i=1}^{n}a_i d_i + \Big(\frac{\kappa}{\gamma}\Big)\Big(\frac{\rho}{\gamma}\Big)^{\kappa}\Big(\frac{1}{\kappa}\Big)\sum_{i=1}^{n}a_i b_i c_i \\
&\quad - \Big(\frac{1}{\gamma}\Big)\Big(\frac{\rho}{\gamma}\Big)^{\kappa}\sum_{i=1}^{n}a_i\log(1-d_i) - \Big(\frac{\kappa}{\gamma}\Big)\Big(\frac{\rho}{\gamma}\Big)^{\kappa}\log\Big(\frac{\rho}{\gamma}\Big)\sum_{i=1}^{n}a_i\log(1-d_i) \\
&= -\Big(\frac{1}{\gamma}\Big)\sum_{i=1}^{n}\delta_i d_i(1+c_i) - \Big(\frac{1}{\gamma}\Big)\Big(\frac{\rho}{\gamma}\Big)^{\kappa}\sum_{i=1}^{n}a_i[d_i+\log(1-d_i)+c_i(d_i-b_i)] \\
&\quad - \Big(\frac{\kappa}{\gamma}\Big)\Big(\frac{\rho}{\gamma}\Big)^{\kappa}\log\Big(\frac{\rho}{\gamma}\Big)\sum_{i=1}^{n}a_i[d_i+\log(1-d_i)].
\end{aligned}
$$

$$\frac{\partial^2 \ell(\boldsymbol{\theta})}{\partial \kappa \partial \rho} = \frac{\partial}{\partial \kappa}\frac{\partial \ell(\boldsymbol{\theta})}{\partial \rho}$$

$$= \frac{m}{\rho} - \Big(\frac{1}{\rho}\Big)\Big(\frac{\rho}{\gamma}\Big)^{\kappa}\sum_{i=1}^{n} a_i \log(1+b_i) - \Big(\frac{\kappa}{\rho}\Big)\Big(\frac{\rho}{\gamma}\Big)^{\kappa}\log\Big(\frac{\rho}{\gamma}\Big)\sum_{i=1}^{n} a_i \log(1+b_i)$$

$$- \Big(\frac{\kappa}{\rho}\Big)\Big(\frac{\rho}{\gamma}\Big)^{\kappa}\Big(\frac{1}{\kappa}\Big)\sum_{i=1}^{n} a_i b_i c_i$$

$$= \frac{m}{\rho} - \Big(\frac{1}{\rho}\Big)\Big(\frac{\rho}{\gamma}\Big)^{\kappa}\sum_{i=1}^{n} a_i[b_i c_i + \log(1+b_i)] - \Big(\frac{\kappa}{\rho}\Big)\Big(\frac{\rho}{\gamma}\Big)^{\kappa}\log\Big(\frac{\rho}{\gamma}\Big)\sum_{i=1}^{n} a_i \log(1+b_i).$$

$$\frac{\partial^2 \ell(\boldsymbol{\theta})}{\partial \kappa \partial \beta_j} = \frac{\partial}{\partial \beta_j}\frac{\partial \ell(\boldsymbol{\theta})}{\partial \kappa} = -\Big(\frac{1}{\kappa}\Big)\Big(\frac{\rho}{\gamma}\Big)^{\kappa}\sum_{i=1}^{n} a_i b_i c_i z_{ij} - \Big(\frac{\rho}{\gamma}\Big)^{\kappa}\log\Big(\frac{\rho}{\gamma}\Big)\sum_{i=1}^{n} a_i \log(1+b_i) z_{ij}.$$

$$\frac{\partial^2 \ell(\boldsymbol{\theta})}{\partial \gamma \partial \rho} = \frac{\partial}{\partial \gamma}\frac{\partial \ell(\boldsymbol{\theta})}{\partial \rho}$$

$$= \Big(\frac{\kappa}{\rho}\Big)\Big(\frac{\kappa}{\gamma}\Big)\Big(\frac{\rho}{\gamma}\Big)^{\kappa}\sum_{i=1}^{n} a_i \log(1+b_i) - \Big(\frac{\kappa}{\rho}\Big)\Big(\frac{\kappa}{\gamma}\Big)\Big(\frac{\rho}{\gamma}\Big)^{\kappa}\sum_{i=1}^{n} a_i d_i$$

$$= -\Big(\frac{\kappa}{\rho}\Big)\Big(\frac{\kappa}{\gamma}\Big)\Big(\frac{\rho}{\gamma}\Big)^{\kappa}\sum_{i=1}^{n} a_i \log(1-d_i) - \Big(\frac{\kappa}{\rho}\Big)\Big(\frac{\kappa}{\gamma}\Big)\Big(\frac{\rho}{\gamma}\Big)^{\kappa}\sum_{i=1}^{n} a_i d_i$$

$$= -\Big(\frac{\kappa}{\rho}\Big)\Big(\frac{\kappa}{\gamma}\Big)\Big(\frac{\rho}{\gamma}\Big)^{\kappa}\sum_{i=1}^{n} a_i[d_i + \log(1-d_i)].$$

.

$$\frac{\partial^2 \ell(\boldsymbol{\theta})}{\partial \gamma \partial \beta_j} = \frac{\partial}{\partial \beta_j}\frac{\partial \ell(\boldsymbol{\theta})}{\partial \gamma}$$

$$= -\Big(\frac{\kappa}{\gamma}\Big)\Big(\frac{\rho}{\gamma}\Big)^{\kappa}\sum_{i=1}^{n} a_i d_i z_{ij} - \Big(\frac{\kappa}{\gamma}\Big)\Big(\frac{\rho}{\gamma}\Big)^{\kappa}\sum_{i=1}^{n} a_i \log(1-d_i) z_{ij}$$

$$= -\Big(\frac{\kappa}{\gamma}\Big)\Big(\frac{\rho}{\gamma}\Big)^{\kappa}\sum_{i=1}^{n} a_i[d_i + \log(1-d_i)] z_{ij}.$$

$$\frac{\partial^2 \ell(\boldsymbol{\theta})}{\partial \rho \partial \beta_j} = \frac{\partial}{\partial \beta_j}\frac{\partial \ell(\boldsymbol{\theta})}{\partial \rho}$$

$$= -\Big(\frac{\kappa}{\rho}\Big)\Big(\frac{\rho}{\gamma}\Big)^{\kappa}\sum_{i=1}^{n} a_i \log(1+b_i) z_{ij}.$$

The maximum likelihood estimate of $\boldsymbol{\theta}^* = (\boldsymbol{\alpha}^{*\prime}, \boldsymbol{\beta}')'$ is obtained by solving the equations $\partial\ell(\boldsymbol{\theta}^*)/\partial\kappa^* = 0$, $\partial\ell(\boldsymbol{\theta}^*)/\partial\gamma^* = 0$, $\partial\ell(\boldsymbol{\theta}^*)/\partial\rho^* = 0$ and $\partial\ell(\boldsymbol{\theta}^*)/\partial\beta_j = 0$ iteratively. The first and second derivatives of $\ell(\boldsymbol{\theta}^*)$ can be derived by noting that

$$\frac{\partial\ell}{\partial\log u} = u\Big(\frac{\partial\ell}{\partial u}\Big), \tag{A.14}$$

$$\frac{\partial^2\ell}{\partial\log u\,\partial\log v} = u\Big(\frac{\partial v}{\partial u}\Big)\Big(\frac{\partial\ell}{\partial v}\Big) + uv\Big(\frac{\partial^2\ell}{\partial u\partial v}\Big), \tag{A.15}$$

$$\frac{\partial^2\ell}{\partial\log u\,\partial v} = u\Big(\frac{\partial^2\ell}{\partial u\partial v}\Big). \tag{A.16}$$

Using (A.14)-(A.16), the first and second derivatives of $\ell(\boldsymbol{\theta}^*)$ can be expressed as

$$\frac{\partial\ell(\boldsymbol{\theta}^*)}{\partial\kappa^*} = \Big[\kappa\Big(\frac{\partial\ell(\boldsymbol{\theta})}{\partial\kappa}\Big)\Big]_{\alpha=e^{\alpha^*}}, \quad \frac{\partial\ell(\boldsymbol{\theta}^*)}{\partial\gamma^*} = \Big[\gamma\Big(\frac{\partial\ell(\boldsymbol{\theta})}{\partial\gamma}\Big)\Big]_{\alpha=e^{\alpha^*}},$$

$$\frac{\partial\ell(\boldsymbol{\theta}^*)}{\partial\rho^*} = \Big[\rho\Big(\frac{\partial\ell(\boldsymbol{\theta})}{\partial\rho}\Big)\Big]_{\alpha=e^{\alpha^*}}, \quad \frac{\partial\ell(\boldsymbol{\theta}^*)}{\partial\beta_j} = \Big[\frac{\partial\ell(\boldsymbol{\theta})}{\partial\beta_j}\Big]_{\alpha=e^{\alpha^*}},$$

$$\frac{\partial^2\ell(\boldsymbol{\theta}^*)}{\partial\kappa^{*2}} = \Big[\kappa\Big(\frac{\partial\ell(\boldsymbol{\theta})}{\partial\kappa}\Big) + \kappa^2\Big(\frac{\partial^2\ell(\boldsymbol{\theta})}{\partial\kappa^2}\Big)\Big]_{\alpha=e^{\alpha^*}},$$

$$\frac{\partial^2\ell(\boldsymbol{\theta}^*)}{\partial\gamma^{*2}} = \Big[\gamma\Big(\frac{\partial\ell(\boldsymbol{\theta})}{\partial\gamma}\Big) + \gamma^2\Big(\frac{\partial^2\ell(\boldsymbol{\theta})}{\partial\gamma^2}\Big)\Big]_{\alpha=e^{\alpha^*}},$$

$$\frac{\partial^2\ell(\boldsymbol{\theta}^*)}{\partial\rho^{*2}} = \Big[\rho\Big(\frac{\partial\ell(\boldsymbol{\theta})}{\partial\rho}\Big) + \rho^2\Big(\frac{\partial^2\ell(\boldsymbol{\theta})}{\partial\rho^2}\Big)\Big]_{\alpha=e^{\alpha^*}},$$

$$\frac{\partial^2\ell(\boldsymbol{\theta}^*)}{\partial\beta_j\partial\beta_{j'}} = \Big[\frac{\partial^2\ell(\boldsymbol{\theta})}{\partial\beta_j\partial\beta_{j'}}\Big]_{\alpha=e^{\alpha^*}}, \quad \frac{\partial^2\ell(\boldsymbol{\theta}^*)}{\partial\kappa^*\partial\gamma^*} = \Big[\kappa\gamma\Big(\frac{\partial^2\ell(\boldsymbol{\theta})}{\partial\kappa\partial\gamma}\Big)\Big]_{\alpha=e^{\alpha^*}},$$

$$\frac{\partial^2\ell(\boldsymbol{\theta}^*)}{\partial\kappa^*\partial\rho^*} = \Big[\kappa\rho\Big(\frac{\partial^2\ell(\boldsymbol{\theta})}{\partial\kappa\partial\rho}\Big)\Big]_{\alpha=e^{\alpha^*}}, \quad \frac{\partial^2\ell(\boldsymbol{\theta}^*)}{\partial\kappa^*\partial\beta_j} = \Big[\kappa\Big(\frac{\partial^2\ell(\boldsymbol{\theta})}{\partial\kappa\partial\beta_j}\Big)\Big]_{\alpha=e^{\alpha^*}},$$

$$\frac{\partial^2\ell(\boldsymbol{\theta}^*)}{\partial\gamma^*\partial\rho^*} = \Big[\gamma\rho\Big(\frac{\partial^2\ell(\boldsymbol{\theta})}{\partial\gamma\partial\rho}\Big)\Big]_{\alpha=e^{\alpha^*}}, \quad \frac{\partial^2\ell(\boldsymbol{\theta}^*)}{\partial\gamma^*\partial\beta_j} = \Big[\gamma\Big(\frac{\partial^2\ell(\boldsymbol{\theta})}{\partial\gamma\partial\beta_j}\Big)\Big]_{\alpha=e^{\alpha^*}},$$

$$\frac{\partial^2\ell(\boldsymbol{\theta}^*)}{\partial\rho^*\partial\beta_j} = \Big[\rho\Big(\frac{\partial^2\ell(\boldsymbol{\theta})}{\partial\rho\partial\beta_j}\Big)\Big]_{\alpha=e^{\alpha^*}}.$$

A.2 Generalized Log-Logistic Model: R Codes

The R codes for the Generalized log-logistic model can be found in the Supplementary Material of our published article (https://link.springer.com/article/10.1186/s40488-016-0054-z/#SupplementaryMaterial)

A.3 MKumW Model: Derivatives of the Log-Likelihood Function

Let $m = \sum_{i=1}^{n} \delta_i$, $a_i = \exp(-t_i^{\kappa})$, $A = a_i t_i^{\kappa} \log t_i (1-a_i)^{\gamma-1} - a_i t_i^{2\kappa} \log t_i (1-a_i)^{\gamma-1} - (\gamma-1) a_i^2 t_i^{2\kappa} (1-a_i)^{\gamma-2}$, and $C = \gamma a_i t_i^{\kappa} \log t_i (1-a_i)^{\gamma-1}$. We have

- $$\frac{\partial}{\partial \kappa} a_i = -a_i (t_i)^{\kappa} \log(t_i), \tag{A.17}$$
- $$\frac{\partial}{\partial \kappa} (1-a_i)^{\gamma} = -\gamma (1-a_i)^{\gamma-1} a_i (t_i)^{\kappa} \log(t_i), \tag{A.18}$$
- $$\frac{\partial}{\partial \kappa} \log(1-a_i) = \frac{a_i (t_i)^{\kappa} \log(t_i)}{1-a_i}, \tag{A.19}$$
- $$\frac{\partial}{\partial \kappa} \log(1-a_i)^{\gamma} = \frac{\gamma a_i (t_i)^{\kappa} \log(t_i)}{1-a_i}, \tag{A.20}$$
- $$\frac{\partial}{\partial \kappa} \log(1-(1-a_i)^{\gamma}) = \frac{\gamma a_i (t_i)^{\kappa} \log(t_i)(1-a_i)^{\gamma-1}}{1-(1-a_i)}, \tag{A.21}$$
- $$\frac{\partial}{\partial \gamma} \log(1-a_i) = (1-a_i) \log(1-a_i), \tag{A.22}$$
- $$\frac{\partial d_i}{\partial \kappa} = \frac{\partial}{\partial \kappa} \frac{a_i}{1+a_i} = \frac{a_i \log a_i}{\kappa(1+a_i)} \Big(1 - \frac{a_i}{1+a_i}\Big) = \frac{c_i d_i}{\kappa}, \tag{A.23}$$
- $$\frac{\partial}{\partial \gamma} \log(1-(1-a_i)^{\gamma}) = \frac{(1-a_i)^{\gamma} \log(1-a_i)}{1-(1-a_i)^{\gamma}}, \tag{A.24}$$
- $$\frac{\partial}{\partial \kappa} ((1-a_i)^{\gamma} \log(1-a_i)) = a_i (t_i)^{\kappa} \log(t_i)(1-a_i)^{\gamma-1} [1 + \gamma \log(1-a_i)]. \tag{A.25}$$

Using (4.15) and (A.17)-(A.25), we can derive the first and second derivatives of the log-likelihood function as follows.

$$\frac{\partial \ell(\theta)}{\partial \kappa} = \frac{m}{\kappa} + \sum_{i=1}^{n} \delta_i \log t_i + (\gamma - 1) \sum_{i=1}^{n} \delta_i \log t_i \frac{a_i t_i^{\kappa}}{(1 - a_i)} - \sum_{i=1}^{n} \delta_i t_i^{\kappa} \log t_i$$
$$+ \sum_{i=1}^{n} \delta_i \gamma \log t_i \frac{a_i t_i^{\kappa} (1 - a_i)^{\gamma - 1}}{(1 - (1 - a_i)^{\gamma})} + \rho \sum_{i=1}^{n} \exp(\mathbf{z}_i' \boldsymbol{\beta}) \gamma \log t_i \frac{a_i t_i^{\kappa} (1 - a_i)^{\gamma - 1}}{(1 - (1 - a_i)^{\gamma})}.$$

$$\frac{\partial \ell(\theta)}{\partial \gamma} = \frac{m}{\gamma} + \sum_{i=1}^{n} \delta_i \log(1 - a_i) - \sum_{i=1}^{n} \delta_i \log(1 - a_i) \frac{(1 - a_i)^{\gamma}}{1 - (1 - a_i)^{\gamma}} + \rho \sum_{i=1}^{n} \exp(\mathbf{z}_i' \boldsymbol{\beta}) \log(1 - a_i) \frac{(1 - a_i)^{\gamma}}{1 - (1 - a_i)^{\gamma}}.$$

$$\frac{\partial l(\theta)}{\partial \rho} = \frac{m}{\rho} + \sum_{i=1}^{n} \exp(\mathbf{z}_i' \boldsymbol{\beta}) \log(1 - (1 - a_i)^{\gamma}).$$

$$\frac{\partial l(\theta)}{\partial \beta_j} = \sum_{i=1}^{n} \delta_i z_{ij}' + \rho \sum_{i=1}^{n} z_{ij}' \exp(\mathbf{z}_i' \boldsymbol{\beta}) \log(1 - (1 - a_i)^{\gamma}) \quad \text{for } j = 1, 2, \ldots, p.$$

$$\frac{\partial^2 \ell(\boldsymbol{\theta})}{\partial \kappa^2} = \frac{\partial}{\partial \kappa} \frac{\partial \ell(\boldsymbol{\theta})}{\partial \kappa}$$
$$= -\frac{m}{\kappa^2} + (\gamma - 1) \sum_{i=1}^{n} \delta_i \log t_i \frac{(1 - a_i)(a_i t_i^{\kappa} \log t_i - t_i^{\kappa} a_i t_i^{\kappa} \log t_i) - a_i^2 t_i^{2\kappa} log t_i}{(1 - a_i)^2}$$
$$\sum_{i=1}^{n} \delta_i t_i^{\kappa} \log^2 t_i + \sum_{i=1}^{n} \delta_i \gamma \log t_i \frac{(1 - (1 - a_i)^{\gamma}) A - a_i t_i^{\kappa} (1 - a_i)^{\gamma - 1} C}{(1 - (1 - a_i)^{\gamma})^2}$$
$$+ \rho \sum_{i=1}^{n} \exp(\mathbf{z}_i' \boldsymbol{\beta}) \gamma \log t_i \frac{(1 - (1 - a_i)^{\gamma}) A - a_i t_i^{\kappa} (1 - a_i)^{\gamma - 1} C}{(1 - (1 - a_i)^{\gamma})^2}.$$

$$\frac{\partial^2 \ell(\boldsymbol{\theta})}{\partial \gamma^2} = \frac{\partial}{\partial \gamma} \frac{\partial \ell(\boldsymbol{\theta})}{\partial \gamma}$$
$$= -\frac{m}{\gamma^2} - \sum_{i=1}^{n} \delta_i \log^2(1 - a_i) \frac{(1 - (1 - a_i)^{\gamma})(1 - a_i)^{\gamma} + (1 - a_i)^{2\gamma}}{(1 - (1 - a_i)^{\gamma})^2}$$
$$+ \rho \sum_{i=1}^{n} \exp(\mathbf{z}_i' \boldsymbol{\beta}) \log^2(1 - a_i) \frac{(1 - (1 - a_i)^{\gamma})(1 - a_i)^{\gamma} + (1 - a_i)^{2\gamma}}{(1 - (1 - a_i)^{\gamma})^2}.$$

$$\frac{\partial^2 \ell(\boldsymbol{\theta})}{\partial \rho^2} = \frac{\partial}{\partial \rho} \frac{\partial \ell(\boldsymbol{\theta})}{\partial \rho} = -\frac{m}{\rho^2}.$$

$$\frac{\partial^2 \ell(\boldsymbol{\theta})}{\partial \beta_j \partial \beta_{j'}} = \frac{\partial}{\partial \beta_j}\frac{\partial \ell(\boldsymbol{\theta})}{\partial \beta_{j'}} = \rho \sum_{i=1}^{n} \exp(\mathbf{z}_i'\boldsymbol{\beta}) \log(1-(1-a_i)^{\gamma}) z_{ij} z_{ij}' \quad \text{for } j, j' = 1, 2, \ldots, p.$$

$$\begin{aligned}
\frac{\partial^2 \ell(\boldsymbol{\theta})}{\partial \kappa \partial \gamma} &= \frac{\partial}{\partial \kappa}\frac{\partial \ell(\boldsymbol{\theta})}{\partial \gamma} \\
&= \sum_{i=1}^{n} \delta_i \frac{a_i t_i^k \log t_i}{1-a_i} - \sum_{i=1}^{n} \delta_i \frac{a_i t_i^k \log t_i (1-b_i)^{\gamma-1}((1-(1-a_i)^{\gamma})(1+\gamma\log(1-a_i)) + (1-a_i)^{\gamma}\log(1-a_i))}{(1-(1-a_i)^{\gamma})^2} \\
&+ \rho \sum_{i=1}^{n} \exp(\mathbf{z}_i'\boldsymbol{\beta}) \frac{a_i t_i^k \log t_i (1-a_i)^{\gamma-1}((1-(1-a_i)^{\gamma})(1+\gamma\log(1-a_i)) + (1-a_i)^{\gamma}\log(1-a_i))}{(1-(1-a_i)^{\gamma})^2}.
\end{aligned}$$

$$\frac{\partial^2 \ell(\boldsymbol{\theta})}{\partial \kappa \partial \rho} = \frac{\partial^2 l(\theta)}{\partial \kappa \partial \rho} = \sum_{i=1}^{n} \exp(\mathbf{z}_i'\boldsymbol{\beta}) \frac{\gamma a_i t_i^k \log t_i (1-a_i)^{\gamma-1}}{1-(1-a_i)^{\gamma}}.$$

$$\begin{aligned}
\frac{\partial^2 \ell(\boldsymbol{\theta})}{\partial \kappa \partial \beta_j} &= \frac{\partial^2 l(\theta)}{\partial \beta \partial \kappa} \\
&= \rho \sum_{i=1}^{n} z_{ij} \exp(\mathbf{z}_i'\boldsymbol{\beta}) \frac{\gamma a_i t_i^k \log t_i (1-a_i)^{\gamma-1}}{1-(1-a_i)^{\gamma}}.
\end{aligned}$$

$$\begin{aligned}
\frac{\partial^2 \ell(\boldsymbol{\theta})}{\partial \rho \partial \beta_j} &= \frac{\partial}{\partial \beta_j}\frac{\partial \ell(\boldsymbol{\theta})}{\partial \rho} \\
&= \sum_{i=1}^{n} z_{ij} \exp(\mathbf{z}_i'\boldsymbol{\beta}) \log(1-(1-a_i)^{\gamma}).
\end{aligned}$$

$$\begin{aligned}
\frac{\partial^2 \ell(\boldsymbol{\theta})}{\partial \gamma \partial \rho} &= \frac{\partial}{\partial \gamma}\frac{\partial \ell(\boldsymbol{\theta})}{\partial \rho} \\
&= \sum_{i=1}^{n} \exp(\mathbf{z}_i'\boldsymbol{\beta}) \frac{(1-a_i)^{\gamma}\log(1-a_i)}{1-(1-a_i)^{\gamma}}.
\end{aligned}$$

$$\begin{aligned}
\frac{\partial^2 \ell(\boldsymbol{\theta})}{\partial \gamma \partial \beta_j} &= \frac{\partial}{\partial \beta_j}\frac{\partial \ell(\boldsymbol{\theta})}{\partial \gamma} \\
&= \rho \sum_{i=1}^{n} z_i^{'} \exp(\mathbf{z}_i'\boldsymbol{\beta}) \frac{(1-a_i)^{\gamma}\log(1-a_i)}{1-(1-a_i)^{\gamma}}.
\end{aligned}$$

A.4 MKumW Model: R Codes to Fit the PH Model

```
library(survival)
library(nlme)
library(gmp)
library(Rmpfr)
library(MASS)
library(numDeriv)
library(JM)
#############################################################
# MKumW PH log(survival probabilities).
# st = survival time.
# kappa, gam and rho = parameters.
# prec.bits = a number, the maximal precision
#     to be used, in bits. This is required
#     for the function mpfr(). The default is NULL,
#     in which case it will R default value.
#############################################################
sEPH<-function(st,kappa,gam,rho,prec.bits=NULL){
  if(!is.null(prec.bits)){
    a0<-mpfr(-(st^kappa),prec.bits) # -t^kappa
  } else{
    a0<--(st^kappa) # -t^kappa
  }
  a<-exp(a0) # a = exp(-t^kappa)
  a2<-log1p(-a) # log(1-a) = log[1 - exp(-t^kappa)]
  b1<-log(-expm1(gam*a2)) #log(1-b) = log(1-(1-a)^gam) = log[1-(1 - exp(-t^
      kappa))^gam]
  s.log<- rho*b1
  return(as.numeric(s.log))
}
#############################################################
hEPH<-function(st,kappa,gam,rho,prec.bits=NULL){
  if(!is.null(prec.bits)){
    a0<-mpfr(-(st^kappa),prec.bits) # -t^kappa
  } else{
    a0<--(st^kappa) # -t^kappa
  }
  a<-exp(a0) # a = exp(-t^kappa)
  a2<-log1p(-a) # log(1-a) = log[1 - exp(-t^kappa)]
  b1<-log(-expm1(gam*a2)) #log(1-b) = log(1-(1-a)^gam) = log[1-(1 - exp(-t^
      kappa))^gam]
  h.log<-log(kappa)+log(gam)+log(rho)+(kappa-1)*log(st)+(gam-1)*a2+a0-b1
  h<-exp(h.log)
  return(as.numeric(h))
}
#############################################################
# Residual plots.
# fit = fit of the model produced using the function
#       fit.model (see below).
```

```
# st = survival time.
# status = censoring indicator.
# xx = covariate matrix.
# model = "MKumWPH" or "WPH" or "cox".
# prec.bits = a number, the maximal precision
#    to be used, in bits. This is required
#    for the function mpfr(). The default is NULL,
#    in which case it will R default value.
##########################################################
diagnostic.plot<-function(fit,st,status,xx=NULL,model,prec.bits=1000){
  if(is.null(xx)&& model=="cox"){
    print("Cox PH is not available for univariate analysis")
    return
  }
  est<-fit$MKumWPH[,1]
  if(!is.null(xx)){
    kappa<-exp(est[ncol(xx)+1])
    gam<-exp(est[ncol(xx)+2])
    rho<-exp(est[ncol(xx)+3])
    beta<-est[1:ncol(xx)]
    pred<-as.matrix(xx)%*%beta
  } else{
    kappa<-exp(est[1])
    gam<-exp(est[2])
    rho<-exp(est[3])
    pred<-rep(0,length(st))
  }
  Rhat<- c(-sEPH(st,kappa,gam,rho,prec.bits=prec.bits)*exp(pred))
  ordered <- order(Rhat)
  Rhat <- Rhat[ordered]
  delta <- status[ordered]
  fitres1<-survfit(Surv(Rhat,delta)~1,type="kaplan-meier")
  rr<--log(fitres1$surv)
  rr[is.infinite(rr)]<-NA
  rr0<-fitres1$time
  rr0[is.na(rr)]<-NA
  rr<-na.omit(rr)
  rr0<-na.omit(rr0)
  est<-fit$WPH[,1]
  if(!is.null(xx)){
    kappa<-exp(est[ncol(xx)+1])
    rho<-exp(est[ncol(xx)+2])
    beta<-est[1:ncol(xx)]
    pred<-as.matrix(xx)%*%beta
  } else{
    kappa<-exp(est[1])
    rho<-exp(est[2])
    pred<-rep(0,length(st))
  }
  Rhat<-(rho*st^kappa)*exp(pred)
  ordered <- order(Rhat)
```

```
Rhat <- Rhat[ordered]
delta <- status[ordered]
fitres1<-survfit(Surv(Rhat,delta)~1,type="kaplan-meier")
rrr<--log(fitres1$surv)
rrr[is.infinite(rrr)]<-NA
rrr0<-fitres1$time
rrr0[is.na(rrr)]<-NA
rrr<-na.omit(rrr)
rrr0<-na.omit(rrr0)
if(!is.null(xx)){
  fit.cox0<-coxph(Surv(st,status)~.,data=xx)
  coxsnellres<-status-resid(fit.cox0,type="martingale")
  fitres<-survfit(coxph(Surv(coxsnellres,status)~1,method='breslow'),type='
      aalen')
  rrrr<--log(fitres$surv)
  rrrr[is.infinite(rrrr)]<-NA
  rrrr0<-fitres$time
  rrrr0[is.na(rrrr0)]<-NA
  rrrr<-na.omit(rrrr)
  rrrr0<-na.omit(rrrr0)
  xy.lim<-max(max(max(rr0,na.rm=TRUE),max(rrr0,na.rm=TRUE),max(rrrr0,na.rm=
      TRUE)),
              max(max(rr,na.rm=TRUE), max(rrr,na.rm=TRUE), max(rrrr,na.rm=
                  TRUE)))+0.05
} else{
  xy.lim<-max(max(max(rr0,na.rm=TRUE),max(rrr0,na.rm=TRUE)),
              max(max(rr,na.rm=TRUE), max(rrr,na.rm=TRUE)))+0.05
}
if(model=="MKumWPH"){
  plot(rr0, rr, type="p", pch=20,
       xlim=c(0,xy.lim),ylim=c(0,xy.lim),
       xlab="Residuals",
       ylab="Estimated Cumulative Hazard Function",main="(a) MKumWPH",
       cex.lab=1.25,cex.main=1.5)
  abline(a=0,b=1)
}
if(model=="WPH"){
  plot(rrr0, rrr, type="p", pch=20,
       xlim=c(0,xy.lim),ylim=c(0,xy.lim),
       xlab="Residuals",
       ylab="Estimated Cumulative Hazard Function",main="(b) Weibull PH",
       cex.lab=1.25,cex.main=1.5)
  abline(a=0,b=1)
}
if(model=="cox"){
  plot(rrrr0,rrrr, type="p", pch=20,
       xlim=c(0,xy.lim),ylim=c(0,xy.lim),
       xlab="Residuals",ylab="Estimated Cumulative Hazard Function",main="(c
           ) Cox PH",
       cex.lab=1.25,cex.main=1.5)
  abline(0,1)
```

```
  }
}
#############################################################
# Generate initial values for MKumWPH.
# This is used in the main function fit.model (see below).
# init = c(tau, eta), initial values to optimize
#    the profile lok-likelihood, given beta
#    (beta is produced from a fit of the Cox model).
#    If init is NULL, default values are used.
# st = survival time.
# status = censoring indicator.
# xx = covariate matrix.
# prec.bits = a number, the maximal precision
#    to be used, in bits. This is required
#    for the function mpfr(). The default is NULL,
#    in which case it will R default value.
#############################################################
MKumWPH.init<-function(init=NULL,st,status,xx=NULL,prec.bits=NULL){
  MKumWPH.llik.init<-function(init,st,status,xx=NULL,beta=NULL,prec.bits=NULL)
      {
    kappa<-exp(init[1])
    gam<-exp(init[2])
    rho<-exp(init[3])
    if(!is.null(xx)){
      pred<-as.matrix(xx)%*%beta
    } else{
      pred<-rep(0,length(st))
    }
    m<-sum(status)
    if(!is.null(prec.bits)){
      a0<-mpfr(-(st^kappa),prec.bits)
    } else{
      a0<--(st^kappa)
    }
    a<-exp(a0) # a
    a1<--expm1(a0) # 1-a
    a2<-log1p(-a) # log(1-a)
    b1<-log(-expm1(gam*a2)) # log(1-(1-a)^gam)
    t1<-status*((kappa-1)*log(st)+(gam-1)*a2+a0-b1+pred)
    t2<-rho*b1*exp(pred)
    lf<-m*log(kappa)+m*log(gam)+m*log(rho)+sum(t1,na.rm=T)+sum(t2,na.rm=T)
    return(-as.numeric(lf))
  }
  if(!is.null(xx)){
    beta<-coef(coxph(Surv(st,status)~.,data=xx))
  }
  if(is.null(init)){
    init1<-rep(1,3)
    init2<-rep(2,3)
    init3<-rep(-1,3)
    init4<-rep(-2,3)
```

```
  init5<-rep(0,3)
}
conv0<-NULL
est0<-NULL
messag<-NULL
options(warn=-1)
fit0<-nlminb(start=init1,objective=MKumWPH.llik.init,st=st,status=status,
             xx=xx,beta=beta,prec.bits=prec.bits)
if(fit0$message=="relative convergence (4)" && is.finite(fit0$objective)){
  conv0<-c(conv0,fit0$objective)
  est0<-rbind(est0,fit0$par)
  messag<-c(messag,fit0$message)
}
fit0<-nlminb(start=init2,objective=MKumWPH.llik.init,st=st,status=status,
             xx=xx,beta=beta,prec.bits=prec.bits)
if(fit0$message=="relative convergence (4)" && is.finite(fit0$objective)){
  conv0<-c(conv0,fit0$objective)
  est0<-rbind(est0,fit0$par)
  messag<-c(messag,fit0$message)
}
fit0<-nlminb(start=init3,objective=MKumWPH.llik.init,st=st,status=status,
             xx=xx,beta=beta,prec.bits=prec.bits)
if(fit0$message=="relative convergence (4)" && is.finite(fit0$objective)){
  conv0<-c(conv0,fit0$objective)
  est0<-rbind(est0,fit0$par)
  messag<-c(messag,fit0$message)
}
fit0<-nlminb(start=init4,objective=MKumWPH.llik.init,st=st,status=status,
             xx=xx,beta=beta,prec.bits=prec.bits)
if(fit0$message=="relative convergence (4)" && is.finite(fit0$objective)){
  conv0<-c(conv0,fit0$objective)
  est0<-rbind(est0,fit0$par)
  messag<-c(messag,fit0$message)
}
fit0<-nlminb(start=init5,objective=MKumWPH.llik.init,st=st,status=status,
             xx=xx,beta=beta,prec.bits=prec.bits)
if(fit0$message=="relative convergence (4)" && is.finite(fit0$objective)){
  conv0<-c(conv0,fit0$objective)
  est0<-rbind(est0,fit0$par)
  messag<-c(messag,fit0$message)
}
if(is.null(messag)){
  return(list(message="non-convergence"))
}
conv00<-min(conv0)
init00<-est0[conv0==conv00,]
if(length(c(init00))>length(fit0$par)){
  init00<-init00[1,]
} else{
  init00<-init00
}
```

```
  message<-messag[conv0==conv00]
  if(length(message)>1){
    message<-message[1]
  }
  if(!is.null(xx)){
    init0<-c(init00,beta)
    names(init0)<-c("logkappa","loggam","logrho",colnames(xx))
    return(list(init=init0,conv=fit0$convergence,obj=conv00,message=message))
  } else{
    names(init00)<-c("logkappa","loggam","logrho")
    return(list(init=init00,conv=fit0$convergence,obj=conv00,message=message))
  }
}
############################################################
# Checking PH assumption using time-dependent
# covariate g(t)
# Only g(t) = log(t) is implemented in this function.
# st = survival time.
# status = censoring indicator.
# xx = covariate matrix.
# fit.cox = Cox fit using the time dependent covariate log(t).
#   For example, if there are two covariates trt and fevc, then
#   fit.cox<-coxph(Surv(st,status)~trt+fevc+tt(trt)+tt(fevc),
#        tt = function(x, t, ...) x * log(t),data=xx)
# length.cut.min = for (0, min(t)), the length of the
#    vector of timepoints to cut at. For example, it can be
#    length.cut.min = 4 for (0, 1).
# length.cut.max = for (0, max(t)), the length of the
#    vector of timepoints to cut at. For example, it can be
#    length.cut.max = 300 for (0, 150).
# prec.bits = a number, the maximal precision
#    to be used, in bits. This is required
#    for the function mpfr(). The default is NULL,
#    in which case it will R default value.
############################################################
############################################################
hint1<-function(vec,point,betat){
  x<-vec[1:point]
  xx<-vec[(point+1):length(vec)]
  return(c(sapply(xx,FUN= function(u) u*log(x))%*%betat))
}
####################################
hEPH.assump<-function(x,kappa,gam,rho,betat,xx,point,prec.bits=NULL){
  mat0<-cbind(x,xx)
  pred<-apply(mat0,1,hint1,point=point,betat=betat)
  if(!is.null(prec.bits)){
    a0<-mpfr(-(x^kappa),prec.bits) # -t^kappa
  } else{
    a0<--(x^kappa) # -t^kappa
  }
  a<-exp(a0) # a = exp(-t^kappa)
```

```
  a2<-log1p(-a) # log(1-a) = log[1 - exp(-t^kappa)]
  b1<-log(-expm1(gam*a2)) #log(1-b) = log(1-(1-a)^gam) = log[1-(1 - exp(-t^
      kappa))^gam]
  h.log<-log(kappa)+log(gam)+log(rho)+(kappa-1)*log(x)+(gam-1)*a2+a0-b1
  h<-exp(t(h.log)+pred)
  if(!is.null(prec.bits)){
    return(asNumeric(h))
  } else{
    return(h)
  }
}
##############################################
gauss_kronrod1<-function(ulim,llim,kappa,gam,rho,
                         betat,xx,prec.bits=NULL){
  t15 <- c(-0.991455371120813, -0.949107912342758, -0.864864423359769,
           -0.741531185599394, -0.586087235467691, -0.405845151377397,
           -0.207784955007899, 0, 0.207784955007899, 0.405845151377397,
           0.586087235467691, 0.741531185599394, 0.864864423359769,
           0.949107912342758, 0.991455371120813)
  c15 <- c(0.0229353220105292, 0.0630920926299785, 0.10479001032225,
           0.140653259715526, 0.169004726639268, 0.190350578064785,
           0.204432940075299, 0.209482141084728, 0.204432940075299,
           0.190350578064785, 0.169004726639268, 0.140653259715526,
           0.10479001032225, 0.0630920926299785, 0.0229353220105292)
  diff.lim<-ulim - llim
  sum.lim<-ulim + llim
  x15<-t(0.5*(sapply(diff.lim,FUN= function(u) u*t15)+
                sapply(sum.lim,FUN= function(u) u*rep(1,15))))
  y15<-hEPH.assump(x15,kappa=kappa,gam=gam,rho=rho,
                   betat=betat,xx=xx,point=15,prec.bits=prec.bits)
  K15 <- colSums(c15 * y15)
  return(K15*diff.lim/2)
}
##############################################
MKumWPH.assump<-function(st,status,xx,fit.cox,prec.bits=NULL){
  ########
  llik.PH.assump0<-function(init,st,status,xx,xx1,beta,betat,prec.bits=NULL){
    kappa<-exp(init[1])
    gam<-exp(init[2])
    rho<-exp(init[3])
    pred0<-c(as.matrix(xx)%*%beta)
    pred1<-c(as.matrix(xx1)%*%betat)
    m<-sum(status)
    if(!is.null(prec.bits)){
      a0<-mpfr(-(st^kappa),prec.bits)
    } else{
      a0<--(st^kappa)
    }
    a<-exp(a0) # a
    a1<--expm1(a0) # 1-a
    a2<-log1p(-a) # log(1-a)
```

```
  b1<-log(-expm1(gam*a2)) # log(1-(1-a)^gam)
  t1<-status*((kappa-1)*log(st)+(gam-1)*a2+a0-b1+pred0+pred1)
  t2<-gauss_kronrod1(ulim=st,llim=rep(0,length(st)),kappa=kappa,gam=gam,rho=
      rho,
                     betat=betat,xx=xx,prec.bits=prec.bits)*exp(pred0)
  lf<-m*log(kappa)+m*log(gam)+m*log(rho)+sum(t1,na.rm=T)-sum(t2,na.rm=T)
  return(-as.numeric(lf))
}
###########
llik.PH.assump<-function(init,st,status,xx,xx1,prec.bits=NULL){
  kappa<-exp(init[1])
  gam<-exp(init[2])
  rho<-exp(init[3])
  beta<-init[4:(4+ncol(xx)-1)]
  betat<-init[(4+ncol(xx)):length(init)]
  pred0<-c(as.matrix(xx)%*%beta)
  pred1<-c(as.matrix(xx1)%*%betat)
  m<-sum(status)
  if(!is.null(prec.bits)){
    a0<-mpfr(-(st^kappa),prec.bits)
  } else{
    a0<--(st^kappa)
  }
  a<-exp(a0) # a
  a1<--expm1(a0) # 1-a
  a2<-log1p(-a) # log(1-a)
  b1<-log(-expm1(gam*a2)) # log(1-(1-a)^gam)
  t1<-status*((kappa-1)*log(st)+(gam-1)*a2+a0-b1+pred0+pred1)
  t2<-gauss_kronrod1(ulim=st,llim=rep(0,length(st)),kappa=kappa,gam=gam,rho=
      rho,
                     betat=betat,xx=xx,prec.bits=prec.bits)*exp(pred0)
  lf<-m*log(kappa)+m*log(gam)+m*log(rho)+sum(t1,na.rm=T)-sum(t2,na.rm=T)
  return(-as.numeric(lf))
}
##################
xx1<-xx*log(st)
cox.est<-coef(fit.cox)
beta<-cox.est[1:ncol(xx)]
betat<-cox.est[(ncol(xx)+1):length(cox.est)]
init<-rep(0,3)
options(warn=-1)
fit0<-nlminb(start=init,objective=llik.PH.assump0,
             st=st,status=status,xx=xx,xx1=xx1,
             beta=beta,betat=betat,prec.bits=prec.bits)
options(warn=0)
if(fit0$message!="relative convergence (4)"){
  init<-rep(-1,3)
  options(warn=-1)
  fit0<-nlminb(start=init,objective=llik.PH.assump0,
               st=st,status=status,xx=xx,xx1=xx1,
               beta=beta,betat=betat,prec.bits=prec.bits)
```

```
  options(warn=0)
}
if(fit0$message!="relative convergence (4)"){
  init<-rep(1,3)
  options(warn=-1)
  fit0<-nlminb(start=init,objective=llik.PH.assump0,
               st=st,status=status,xx=xx,xx1=xx1,
               beta=beta,betat=betat,prec.bits=prec.bits)
  options(warn=0)
}
#######
init<-c(fit0$par,cox.est)
options(warn=-1)
fit1<-nlminb(start=init,objective=llik.PH.assump,
             st=st,status=status,xx=xx,xx1=xx1,prec.bits=prec.bits)
options(warn=0)
if(fit1$message!="relative convergence (4)"){
  init<-c(rep(0,3),cox.est)
  options(warn=-1)
  fit1<-nlminb(start=init,objective=llik.PH.assump,
               st=st,status=status,xx=xx,xx1=xx1,prec.bits=prec.bits)
  options(warn=0)
}
if(fit1$message!="relative convergence (4)"){
  init<-c(rep(1,3),cox.est)
  options(warn=-1)
  fit1<-nlminb(start=init,objective=llik.PH.assump,
               st=st,status=status,xx=xx,xx1=xx1,prec.bits=prec.bits)
  options(warn=0)
}
if(fit1$message!="relative convergence (4)"){
  init<-c(rep(-1,3),cox.est)
  options(warn=-1)
  fit1<-nlminb(start=init,objective=llik.PH.assump,
               st=st,status=status,xx=xx,xx1=xx1,prec.bits=prec.bits)
  options(warn=0)
}
#############
#hess<-fdHess(pars=fit1$par,fun=llik.PH.assump,st=st,status=status,xx=xx,xx1
    =xx1)$Hessian
#if(is.na(sum(hess))){
hess<-hessian(func=llik.PH.assump,x=fit1$par,st=st,status=status,xx=xx,xx1=
    xx1)
#}
if(all(eigen(hess)$values>0) && !is.na(sum(hess))){
  cov.mat<-solve(hess)
} else{
  warning("MKumWPH covariance matrix is not positive definite")
}
##############
se<-sqrt(diag(cov.mat))
```

```
  z<-fit1$par/se
  p.val<-round(2*pnorm(abs(z),lower.tail=F),digits=4)
  MKumWPH.results0<-cbind(est=fit1$par,se=se,z=z,p=p.val)
  rownames(MKumWPH.results0)[1:3]<-c("logkappa","loggam","logrho")
  MKumWPH.results<-rbind(MKumWPH.results0[4:nrow(MKumWPH.results0),],MKumWPH.
      results0[1:3,])
  ################
  fit.cox0<-coxph(Surv(st,status)~.,data=xx)
  zp <- cox.zph(fit.cox0, transform= function(time) log(time))
  if(ncol(xx)<=3){
    par(mfrow=c(1,ncol(xx)))
  } else{
    par(mfrow=c(ceiling(ncol(xx)/3),3))
  }
  for(i in 1:ncol(xx)){
    plot(zp[i],xlab="",ylab="")
    abline(MKumWPH.results[c(i,(ncol(xx)+i)),1],col="gray75")
    abline(coef(fit.cox)[c(i,(ncol(xx)+i))],col="gray75",lty=2)
  }
  return(list(MKumWPH.results=MKumWPH.results,convergence=fit1$message,cox.
      results=fit.cox))
}
#########################################################
# Fit the MKumWPH, Weibull PH and Cox models.
# init = c(tau, eta, rho, beta). If NULL, MKumWPH.init (see above)
#    is used to generate initial values.
# st = survival time.
# status = censoring indicator.
# xx = covariate matrix.
# conf.level = confidence level (default is 0.95).
# prec.bits = a number, the maximal precision
#    to be used, in bits. This is required
#    for the function mpfr(). The default is 500.
#########################################################
fit.model<-function(init=NULL,st,status,xx=NULL,conf.level=0.95,MKumWPH.cov=
    FALSE,dat=FALSE){
  MKumWPH.llik<-function(init,st,status,xx=NULL,prec.bits=NULL){
    kappa<-exp(init[1])
    gam<-exp(init[2])
    rho<-exp(init[3])
    if(!is.null(xx)){
      beta<-init[4:length(init)]
      pred<-as.matrix(xx)%*%beta
    } else{
      pred<-rep(0,length(st))
    }
    m<-sum(status)
    if(!is.null(prec.bits)){
      a0<-mpfr(-(st^kappa),prec.bits)
    } else{
      a0<--(st^kappa)
```

```
  }
  a<-exp(a0) # a
  a1<--expm1(a0) # 1-a
  a2<-log1p(-a) # log(1-a)
  b1<-log(-expm1(gam*a2)) # log(1-(1-a)^gam)
  t1<-status*((kappa-1)*log(st)+(gam-1)*a2+a0-b1+pred)
  t2<-rho*b1*exp(pred)
  lf<-m*log(kappa)+m*log(gam)+m*log(rho)+sum(t1,na.rm=T)+sum(t2,na.rm=T)
  return(-as.numeric(lf))
}
WPH.llik<-function(init,st,status,xx=NULL){
  kappa<-exp(init[1])
  rho<-exp(init[2])
  m<-sum(status)
  if(!is.null(xx)){
    beta<-init[3:length(init)]
    pred<-as.matrix(xx)%*%beta
  } else{
    pred<-rep(0,length(st))
  }
  t1<-status*((kappa-1)*log(st)+pred)
  t2<- rho*(st^kappa)*exp(pred)
  lf<-m*log(kappa)+m*log(rho)+sum(t1,na.rm=T)-sum(t2,na.rm=T)
  return(-lf)
}
c.value<-abs(qnorm((1-conf.level)/2))
if(is.null(init)){
  init00<-MKumWPH.init(st=st,status=status,xx=xx)
  if(init00$message!="relative convergence (4)"){
    init00<-MKumWPH.init(st=st,status=status,xx=xx,prec.bits=500)
  }
  init<-init00$init
}
options(warn=-1)
fit<-nlminb(start=init,objective=MKumWPH.llik,st=st,status=status,xx=xx)

if(fit$message!="relative convergence (4)" || !is.finite(fit$objective)){
  fit<-nlminb(start=rep(0,(3+ncol(xx))),objective=MKumWPH.llik,st=st,status=
      status,xx=xx)
}
if(fit$message!="relative convergence (4)" || !is.finite(fit$objective)){
  fit<-nlminb(start=rep(-1,(3+ncol(xx))),objective=MKumWPH.llik,st=st,status
      =status,xx=xx)
}
if(fit$message!="relative convergence (4)" || !is.finite(fit$objective)){
  fit<-nlminb(start=rep(1,(3+ncol(xx))),objective=MKumWPH.llik,st=st,status=
      status,xx=xx)
}
if(fit$message!="relative convergence (4)" || !is.finite(fit$objective)){
  fit<-nlminb(start=init,objective=MKumWPH.llik,st=st,status=status,xx=xx,
      prec.bits=500)
```

```
}
if(fit$message!="relative convergence (4)" || !is.finite(fit$objective)){
  fit<-nlminb(start=rep(0,(3+ncol(xx))),objective=MKumWPH.llik,st=st,status=
      status,xx=xx,prec.bits=500)
}
options(warn=0)
est<-fit$par
mllik<- -fit$objective
conv<-fit$message
inf.mat0<-hessian(MKumWPH.llik,x=est,st=st,status=status,xx=xx)
if(is.na(sum(inf.mat0))){
  inf.mat0<-hessian(MKumWPH.llik,x=est,st=st,status=status,xx=xx,prec.bits
      =500)
}
if(all(eigen(inf.mat0)$values>0)){
  cov.mat<-solve(inf.mat0)
} else{
  warning("MKumWPH covariance matrix is not positive definite")
}
se<-sqrt(diag(cov.mat))
lower<-est-c.value*se
upper<-est+c.value*se
z<-est/se
p.val<-2*pnorm(abs(z),lower.tail=F)
aic<--2*mllik+2*(length(est))
results<-cbind(est=est,se=se,lower.CI=lower,upper.CI=upper,z=z,p=p.val)
colnames(results)<-c("est","se","lower.CI","upper.CI","z","p")
if(!is.null(xx)){
  results<-data.frame(rbind(results[4:nrow(results),],results[1:3,]))
  fit.cox<-coxph(Surv(st,status)~.,data=xx)
  sum.cox<-as.matrix(summary(fit.cox)$coef)
  est.cox<-sum.cox[,1]
  se.cox<-sum.cox[,3]
  lower.cox<-est.cox-c.value*se.cox
  upper.cox<-est.cox+c.value*se.cox
  z.cox<-sum.cox[,4]
  p.cox<-sum.cox[,5]
  results.cox<-data.frame(est=est.cox,se=se.cox,
                          lower.CI=lower.cox,upper.CI=upper.cox,z=z.cox,p=p.
                              cox)
  rownames(results.cox)<-rownames(sum.cox)
  aic.cox<-extractAIC(fit.cox)[2]
  cox.maxllik<-fit.cox$loglik[2]
  fit.w<-survreg(Surv(st,status)~.,dist="weibull",data=xx)
  est0.w<-fit.w$coef
  log.kappa.w<-log(1/fit.w$scale)
  log.rho.w<- -est0.w[1]*exp(log.kappa.w)
  beta.w<- -est0.w[2:length(est0.w)]*exp(log.kappa.w)
  est.w<-c(log.kappa.w,log.rho.w,beta.w)
  mllik.w<-fit.w$loglik[2]
  aic.w<-extractAIC(fit.w)[2]
```

```
  inf.mat.w<-hessian(func=WPH.llik,x=est.w,st=st,status=status,xx=xx)
  cov.mat.w<-solve(inf.mat.w)
  se.w<-sqrt(diag(cov.mat.w))
  z.w<-est.w/se.w
  p.val0.w<-2*pnorm(-abs(z.w))
  lower.w<-est.w-se.w*c.value
  upper.w<-est.w+se.w*c.value
  results.w<-cbind(est=est.w,se=se.w,lower.CI=lower.w,upper.CI=upper.w,z=z.w
      ,p=p.val0.w)
  results.w<-rbind(results.w[3:nrow(results.w),],results.w[1:2,])
  rownames(results.w)<-c(rownames(sum.cox),"logkappa","logrho")
  rownames(results)<-c(rownames(sum.cox),"logkappa","loggam","logrho")
  lr.test<-2*(mllik-mllik.w)
  p.lr.test<-pchisq(lr.test,1,lower.tail=FALSE)
  LR.test<-cbind(chi.sq=lr.test,p.chi.sq=p.lr.test)
}
if(is.null(xx)){
  fit.w<-survreg(Surv(st,status)~1,dist="weibull")
  est0.w<-fit.w$coef
  log.kappa.w<-log(1/fit.w$scale)
  log.rho.w<- -est0.w[1]*exp(log.kappa.w)
  est.w<-c(log.kappa.w,log.rho.w)
  mllik.w<-fit.w$loglik[2]
  aic.w<-extractAIC(fit.w)[2]
  inf.mat.w<-hessian(func=WPH.llik,x=est.w,st=st,status=status,xx=xx)
  cov.mat.w<-solve(inf.mat.w)
  se.w<-sqrt(diag(cov.mat.w))
  z.w<-est.w/se.w
  p.val0.w<-2*pnorm(-abs(z.w))
  lower.w<-est.w-se.w*c.value
  upper.w<-est.w+se.w*c.value
  results.w<-cbind(est=est.w,se=se.w,lower.CI=lower.w,upper.CI=upper.w,z=z.w
      ,p=p.val0.w)
  rownames(results.w)<-c("logkappa","logrho")
  lr.test<-2*(mllik-mllik.w)
  p.lr.test<-pchisq(lr.test,1,lower.tail=FALSE)
  LR.test<-cbind(chi.sq=lr.test,p.chi.sq=p.lr.test)
  if(MKumWPH.cov){
    final.res<-list(MKumWPH=results,MKumWPH.mllik=mllik,MKumWPH.AIC=aic,cov.
        mat=cov.mat,convergence=conv,
                    WPH=results.w,WPH.mllik=mllik.w,WPH.AIC=aic.w,EW.vs.W.LR
                        .Test=LR.test)
  } else{
    final.res<-list(MKumWPH=results,MKumWPH.mllik=mllik,MKumWPH.AIC=aic,
        convergence=conv,
                    WPH=results.w,WPH.mllik=mllik.w,WPH.AIC=aic.w,EW.vs.W.LR
                        .Test=LR.test)
  }

  if(dat){
    dat.mat<-data.frame(st,status)
```

```
      final.res[["data"]]<-dat.mat
    }
  } else{
    if(MKumWPH.cov){
      final.res<-list(MKumWPH=results,MKumWPH.mllik=mllik,MKumWPH.AIC=aic,cov.
          mat=cov.mat,convergence=conv,
                      WPH=results.w,WPH.mllik=mllik.w,WPH.AIC=aic.w,EW.vs.W.LR
                          .Test=LR.test,
                      cox=results.cox,cox.mllik=cox.maxllik,cox.AIC=aic.cox)
    } else{
      final.res<-list(MKumWPH=results,MKumWPH.mllik=mllik,MKumWPH.AIC=aic,
          convergence=conv,
                      WPH=results.w,WPH.mllik=mllik.w,WPH.AIC=aic.w,EW.vs.W.LR
                          .Test=LR.test,
                      cox=results.cox,cox.mllik=cox.maxllik,cox.AIC=aic.cox)
    }
    if(dat){
      dat.mat<-data.frame(st,status,xx)
      final.res[["data"]]<-dat.mat
    }
  }
  return(final.res)
}
############################################################
# Analysis of rhDNase data.
# Data are given in Prof. R.J. Cook's website
# (http://www.math.uwaterloo.ca/~rjcook/cook-lawless-recurrent_code.html).
# Data download link:
# http://www.math.uwaterloo.ca/~rjcook/cook-lawless-recurrent/example2/rhDNase
    .dat

############################################################
dat<-read.table("rhDNase.txt",header=TRUE)
fevc<-dat$fev-mean(dat$fev)
dat<-data.frame(dat,fevc=fevc)
xx<-data.frame(trt=dat$trt,fevc=dat$fevc)
st<-dat$time
status<-dat$status
fit<-fit.model(st=st,status=status,xx=xx,MKumWPH.cov=FALSE)


round(fit$MKumWPH, digits=3)
round(fit$WPH, digits=3)
round(fit$cox, digits=3)

kappa<-exp(fit$MKumWPH[3,1])
gam<-exp(fit$MKumWPH[4,1])
rho<-exp(fit$MKumWPH[5,1])
hh<-hEPH(st,kappa=kappa,gam=gam,rho=rho)
ord<-order(st)
st0<-st[ord]
```

```
hh0<-hh[ord]
plot(st0,hh0,type="l",lty=1,xlab="t",ylab="h(t)",main="Hazard function")

par(mfrow=c(1,3))
diagnostic.plot(fit,st,status,xx=xx,model="MKumWPH",prec.bits=1000)
diagnostic.plot(fit,st,status,xx=xx,model="WPH",prec.bits=1000)
diagnostic.plot(fit,st,status,xx=xx,model="cox",prec.bits=1000)

fit.cox<-coxph(Surv(st,status)~trt+fevc+tt(trt)+tt(fevc),
               tt = function(x, t, ...) x * log(t),data=xx)

PH.assum<-MKumWPH.assump(st=st,status=status,xx=xx,fit.cox=fit.cox,prec.bits=
    NULL)

round(PH.assum$MKumWPH.results,digits=3)
round(summary(fit.cox)$coef,digits=3)

#########################################################################
```

A.5 MKumW Model: R Codes to Fit the Recurrent Event Model

```
library(survival)
library(Rmpfr)
library(numDeriv)
library(nlme)
#################################################
#################################################
# Note: For recurrent data analysis, data must be
#   in counting process format. For detail, see
#   "The Statistical Analysis of Recurrent Events"
#   by Cook and Lawless.
#################################################

#################################################
# Fit the MKumWPH, Weibull PH and Andersen-Gill models.
# init.MKumWPH = c(tau, eta, rho, beta). If NULL,
#     MKumWPH.recurrent.init (see above)
#     is used to generate initial values.
# init.WPH = c(tau, rho, beta). If NULL,
#     WPH.recurrent.init (see above)
#     is used to generate initial values.
# begin = start time for each interval of follow-up.
# stop = stop time for each interval of follow-up.
# status = event status (failure or censored).
# xx = covariate matrix.
# conf.level = confidence level (default is 0.95).
# prec.bits = a number, the maximal precision
#    to be used, in bits. This is required
#    for the function mpfr(). The default is NULL,
#    in which case it will use R default value.
```

```
#################################################
fit.recurrent.model<-function(init.MKumWPH=NULL, init.WPH=NULL, begin, stop,
    status, xx,
                                        conf.level=0.95, prec.bits=NULL){
  MKumWPH.recurrent.llik<-function(init.MKumWPH, begin, stop, status, xx, prec.bits
      =NULL){
    kappa<-exp(init.MKumWPH[1])
    gam<-exp(init.MKumWPH[2])
    rho<-exp(init.MKumWPH[3])
    if(!is.null(xx)){
      beta<-init.MKumWPH[4:length(init.MKumWPH)]
      pred<-as.matrix(xx)%*%beta
    } else{
      pred<-rep(0,length(stop))
    }
    m<-sum(status)
    if(!is.null(prec.bits)){
      a0<-mpfr(-(stop^kappa),prec.bits)
      b0<-mpfr(-(begin^kappa),prec.bits)
    } else{
      a0<--(stop^kappa)
      b0<--(begin^kappa)
    }
    a<-exp(a0) # a
    a2<-log1p(-a) # log(1-a)
    aa<-log(-expm1(gam*a2)) #log(1-(1-a)^gam)
    b<-exp(b0)
    b2<-log1p(-b)
    bb<-log(-expm1(gam*b2))
    t1<-status*((kappa-1)*log(stop)+(gam-1)*a2+a0-aa+pred)
    t2<- -rho*(aa-bb)*exp(pred)
    lf<-m*log(kappa)+m*log(gam)+m*log(rho)+sum(t1,na.rm=T)-sum(t2,na.rm=T)
    return(-as.numeric(lf))
  }
  WPH.recurrent.llik<-function(init.WPH, begin, stop, status, xx){
    kappa<-exp(init.WPH[1])
    rho<-exp(init.WPH[2])
    if(!is.null(xx)){
      beta<-init.WPH[3:length(init.WPH)]
      pred<-as.matrix(xx)%*%beta
    } else{
      pred<-rep(0,length(stop))
    }
    m<-sum(status)
    t1<-status*((kappa-1)*log(stop)+pred)
    t2<- rho*(stop^kappa-begin^kappa)*exp(pred)
    lf<-m*log(kappa)+m*log(rho)+sum(t1,na.rm=T)-sum(t2,na.rm=T)
    return(-as.numeric(lf))
  }
  c.value<-abs(qnorm((1-conf.level)/2))
  if(is.null(init.WPH)){
```

```
  init00<-WPH.recurrent.init(begin=begin,stop=stop,status=status,xx=xx)
  init.WPH<-init00$init
}
options(warn=-1)
fit.w<-nlminb(start=init.WPH,objective=WPH.recurrent.llik,
              begin=begin,stop=stop,status=status,xx=xx)
if(fit.w$message!="relative convergence (4)"){
  iinit<-rep(0,(2+ncol(xx)))
  fit.w<-nlminb(start=iinit,objective=WPH.recurrent.llik,
                begin=begin,stop=stop,status=status,xx=xx)
}
options(warn=0)
est.w<-fit.w$par
mllik.w<- -fit.w$objective
conv.w<-fit.w$message
inf.mat.w<-hessian(func=WPH.recurrent.llik,x=est.w,begin=begin,stop=stop,
                   status=status,xx=xx)
cov.mat.w<-solve(inf.mat.w)
se.w<-sqrt(diag(cov.mat.w))
lower.w<-est.w-c.value*se.w
upper.w<-est.w+c.value*se.w
z.w<-est.w/se.w
p.val.w<-2*pnorm(abs(z.w),lower.tail=F)
aic.w<--2*mllik.w+2*(length(est.w))
results.w<-cbind(est=est.w,se=se.w,lower.CI=lower.w,upper.CI=upper.w,z=z.w,p
    =p.val.w)
rownames(results.w)<-c("logkappa","logrho",colnames(xx))
results.w<-data.frame(rbind(results.w[3:nrow(results.w),],results.w[1:2,]))
if(is.null(init.MKumWPH)){
  init00<-MKumWPH.recurrent.init(begin=begin,stop=stop,status=status,xx=xx,
      prec.bits=prec.bits)
  if(init00$message!="relative convergence (4)"){
    init00<-MKumWPH.recurrent.init(begin=begin,stop=stop,status=status,xx=xx
        ,prec.bits=500)
  }
  init.MKumWPH<-init00$init
}
options(warn=-1)
fit<-nlminb(start=init.MKumWPH,objective=MKumWPH.recurrent.llik,
            begin=begin,stop=stop,status=status,xx=xx)
if(fit$message!="relative convergence (4)"){
  iinit<-rep(0,(3+ncol(xx)))
  fit<-nlminb(start=iinit,objective=MKumWPH.recurrent.llik,
              begin=begin,stop=stop,status=status,xx=xx)
}
if(fit$message!="relative convergence (4)"){
  iinit<-rep(1,(3+ncol(xx)))
  fit<-nlminb(start=iinit,objective=MKumWPH.recurrent.llik,
              begin=begin,stop=stop,status=status,xx=xx)
}
if(fit$message!="relative convergence (4)"){
```

```
  iinit<-rep(-1,(3+ncol(xx)))
  fit<-nlminb(start=iinit,objective=MKumWPH.recurrent.llik,
              begin=begin,stop=stop,status=status,xx=xx)
}
if(fit$message!="relative convergence (4)"){
  fit<-nlminb(start=init.MKumWPH,objective=MKumWPH.recurrent.llik,
              begin=begin,stop=stop,status=status,xx=xx,prec.bits=500)
}
if(fit$message!="relative convergence (4)"){
  iinit<-rep(0,(3+ncol(xx)))
  fit<-nlminb(start=iinit,objective=MKumWPH.recurrent.llik,
              begin=begin,stop=stop,status=status,xx=xx,prec.bits=500)
}
if(fit$message!="relative convergence (4)"){
  return("MKumWPH non-convergence")
}
options(warn=0)
est<-fit$par
mllik<- -fit$objective
conv<-fit$message
inf.mat<-hessian(func=MKumWPH.recurrent.llik,x=est,begin=begin,stop=stop,
                 status=status,xx=xx)
if(is.na(sum(inf.mat))){
  inf.mat<-hessian(func=MKumWPH.recurrent.llik,x=est,begin=begin,stop=stop,
                   status=status,xx=xx,prec.bits=500)
}
if(is.na(sum(inf.mat)) || !all(eigen(inf.mat)$values>0)){
  warning("MKumWPH information is not positive definite")
}
cov.mat<-solve(inf.mat)
se<-sqrt(diag(cov.mat))
lower<-est-c.value*se
upper<-est+c.value*se
z<-est/se
p.val<-2*pnorm(abs(z),lower.tail=F)
aic<--2*mllik+2*(length(est))
results<-cbind(est=est,se=se,lower.CI=lower,upper.CI=upper,z=z,p=p.val)
colnames(results)<-c("est","se","lower.CI","upper.CI","z","p")
rownames(results)<-c("logkappa","loggam","logrho",colnames(xx))
results<-data.frame(rbind(results[4:nrow(results),],results[1:3,]))
lr.test<-2*(mllik-mllik.w)
p.lr.test<-pchisq(lr.test,1,lower.tail=FALSE)
LR.test<-cbind(chi.sq=lr.test,p.chi.sq=p.lr.test)
cox.fit<-coxph(Surv(begin,stop,status)~.,data=xx)
sum.cox<-as.matrix(summary(cox.fit)$coef)
est.cox<-sum.cox[,1]
se.cox<-sum.cox[,3]
lower.cox<-est.cox-c.value*se.cox
upper.cox<-est.cox+c.value*se.cox
z.cox<-sum.cox[,4]
p.cox<-sum.cox[,5]
```

```
  results.cox<-data.frame(est=est.cox,se=se.cox,
                          lower.CI=lower.cox,upper.CI=upper.cox,z=z.cox,p=p.
                              cox)
  rownames(results.cox)<-colnames(xx)
  aic.cox<-extractAIC(cox.fit)[2]
  cox.maxllik<-cox.fit$loglik[2]
  rownames(results)[1:ncol(xx)]<-rownames(results.cox)

  rownames(results.w)[1:ncol(xx)]<-rownames(results.cox)
  return(list(MKumWPH=results,MKumWPH.mllik=mllik,MKumWPH.AIC=aic,MKumWPH.
      convergence=conv,
              WPH=results.w,WPH.mllik=mllik.w,WPH.AIC=aic.w,WPH.convergence=
                  conv.w,EW.vs.W.LR.Test=LR.test,
              cox=results.cox,cox.mllik=cox.maxllik,cox.AIC=aic.cox))
}

#####################################################
#Residual plots.
# fit = fit of the model produced using the function
#       fit.recurrent.model.
# begin = start time for each interval of follow-up.
# stop = stop time for each interval of follow-up.
# status = event status (failure or censored).
# xx = covariate matrix.
# id = subject id
# model = "MKumWPH" or "WPH" or "AG".

#####################################################################
recurrent.diagnostic.plot<-function(fit,begin,stop,status,xx,id,model,prec.
    bits=NULL){
  est.EW<-fit$MKumWPH[,1]
  kappa<-exp(est.EW[ncol(xx)+1])
  gam<-exp(est.EW[ncol(xx)+2])
  rho<-exp(est.EW[ncol(xx)+3])
  beta<-est.EW[1:ncol(xx)]
  pred<-as.matrix(xx)%*%beta
  if(!is.null(prec.bits)){
    a0<-mpfr(-(stop^kappa),prec.bits)
    b0<-mpfr(-(begin^kappa),prec.bits)
  } else{
    a0<--(stop^kappa)
    b0<--(begin^kappa)
  }
  a<-exp(a0) # a
  a2<-log1p(-a) # log(1-a)
  aa<-log(-expm1(gam*a2)) #log(1-b) = log(1-(1-a)^gam)
  b<-exp(b0)
  b2<-log1p(-b)
  bb<-log(-expm1(gam*b2))
  Rhat1<- c(-rho*bb*exp(pred))
  Rhat2<- c(-rho*aa*exp(pred))
```

```
Rhat0<- Rhat2-Rhat1
stop0<-as.vector(unlist(tapply(Rhat0,id,cumsum)))
begin0<-as.vector(unlist(tapply(stop0,id,function(x){if(length(x)==1) {0}
    else {c(0,x[1:(length(x)-1)])}})))
dat0<-data.frame(id,begin0,stop0,status)
dat1<-dat0[(dat0$begin0<dat0$stop0),]
fit0<-survfit(Surv(begin0,stop0,status)~1,data=dat1)
rr<--log(fit0$surv)
rr[is.infinite(rr)]<-NA
rr0<-fit0$time
rr0[is.na(rr)]<-NA
rr<-na.omit(rr)
rr0<-na.omit(rr0)
#########
est.w<-fit$WPH[,1]
kappa.w<-exp(est.w[ncol(xx)+1])
rho.w<-exp(est.w[ncol(xx)+2])
beta.w<-est.w[1:ncol(xx)]
pred.w<-as.matrix(xx)%*%beta.w
Rhat1.w<-c((rho.w*begin^kappa.w)*exp(pred.w))
Rhat2.w<-c((rho.w*stop^kappa.w)*exp(pred.w))
Rhat0.w<- Rhat2.w-Rhat1.w
stop0<-as.vector(unlist(tapply(Rhat0.w,id,cumsum)))
begin0<-as.vector(unlist(tapply(stop0,id,function(x){if(length(x)==1) {0}
    else {c(0,x[1:(length(x)-1)])}})))
dat0<-data.frame(id,begin0,stop0,status)
dat1<-dat0[(dat0$begin0<dat0$stop0),]
fit0<-survfit(Surv(begin0,stop0,status)~1,data=dat1)
rrr<--log(fit0$surv)
rrr[is.infinite(rrr)]<-NA
rrr0<-fit0$time
rrr0[is.na(rrr)]<-NA
rrr<-na.omit(rrr)
rrr0<-na.omit(rrr0)
################
fit.c<-coxph(Surv(begin,stop,status)~.,data=xx)
Rhat0.c<-status-resid(fit.c,type="martingale")
stop0<-as.vector(unlist(tapply(Rhat0.c,id,cumsum)))
begin0<-as.vector(unlist(tapply(stop0,id,function(x){if(length(x)==1) {0}
    else {c(0,x[1:(length(x)-1)])}})))
dat0<-data.frame(id,begin0,stop0,status)
dat1<-dat0[(dat0$begin0<dat0$stop0),]
fit0<-survfit(Surv(begin0,stop0,status)~1,data=dat1)
rrrr<--log(fit0$surv)
rrrr[is.infinite(rrrr)]<-NA
rrrr0<-fit0$time
rrrr0[is.na(rrrr)]<-NA
rrrr<-na.omit(rrrr)
rrrr0<-na.omit(rrrr0)
#########################################################################
xy.lim<-max(max(max(rr0,na.rm=TRUE),max(rrr0,na.rm=TRUE),max(rrrr0,na.rm=
```

```
      TRUE)),
                max(max(rr,na.rm=TRUE), max(rrr,na.rm=TRUE), max(rrrr,na.rm=TRUE
                    )))+0.05
  if(model=="MKumWPH"){
    plot(rr0, rr, type="p", pch=20,
         xlim=c(0,xy.lim),ylim=c(0,xy.lim),
         xlab="Residuals",
         ylab="Estimated Cumulative Hazard",main="(a) MKW PH",
         cex.lab=1.25,cex.main=1.5)
    abline(a=0,b=1)
  }
  if(model=="WPH"){
    plot(rrr0, rrr, type="p", pch=20,
         xlim=c(0,xy.lim),ylim=c(0,xy.lim),
         xlab="Residuals",
         ylab="Estimated Cumulative Hazard",main="(b) Weibull PH",
         cex.lab=1.25,cex.main=1.5)
    abline(a=0,b=1)
  }
  if(model=="AG"){
    plot(rrrr0,rrrr, type="p", pch=20,
         xlim=c(0,xy.lim),ylim=c(0,xy.lim),
         xlab="Residuals",ylab="Estimated Cumulative Hazard Function",main="(c
             ) Andersen-Gill Model",
         cex.lab=1.25,cex.main=1.5)
    abline(0,1)
  }
}
#####################################################
# Checking PH assumption using time-dependent
# covariate g(t).
# Only g(t) = log(t) is implemented in this function.
# id = subject id
# begin = start time for each interval of follow-up.
# stop = stop time for each interval of follow-up.
# status = event status (failure or censored).
# xx = covariate matrix.
# fit.cox = Cox fit using the time dependent covariate log(t).
#  For example, if there are three covariates
#  tx, num and size, then
#   fit.cox<-coxph(Surv(begin,stop,status)~tx+num+size+tt(tx)+
#    tt(num)+tt(size),tt = function(x, t, ...) x * log(t),data=xx)
# length.cut.min = for (0, min(t)), the length of the
#    vector of timepoints to cut at. For example, it can be
#    length.cut.min = 4 for (0, 1).
# length.cut.max = for (0, max(t)), the length of the
#    vector of timepoints to cut at. For example, it can be
#    length.cut.max = 300 for (0, 150).
#########################################################
hint1<-function(vec,point,betat){
  x<-vec[1:point]
```

```
  xx<-vec[(point+1):length(vec)]
  return(c(sapply(xx,FUN= function(u) u*log(x))%*%betat))
}
####################################
hEPH.assump<-function(x,kappa,gam,rho,betat,xx,point,prec.bits=NULL){
  mat0<-cbind(x,xx)
  pred<-apply(mat0,1,hint1,point=point,betat=betat)
  if(!is.null(prec.bits)){
    a0<-mpfr(-(x^kappa),prec.bits) # -t^kappa
  } else{
    a0<--(x^kappa) # -t^kappa
  }
  a<-exp(a0) # a = exp(-t^kappa)
  a2<-log1p(-a) # log(1-a) = log[1 - exp(-t^kappa)]
  b1<-log(-expm1(gam*a2)) #log(1-b) = log(1-(1-a)^gam) = log[1-(1 - exp(-t^
      kappa))^gam]
  h.log<-log(kappa)+log(gam)+log(rho)+(kappa-1)*log(x)+(gam-1)*a2+a0-b1
  h<-exp(t(h.log)+pred)
  if(!is.null(prec.bits)){
    return(asNumeric(h))
  } else{
    return(h)
  }
}
##############################################
gauss_kronrod1<-function(ulim,llim,kappa,gam,rho,
                         betat,xx,prec.bits=NULL){
  t15 <- c(-0.991455371120813, -0.949107912342758, -0.864864423359769,
           -0.741531185599394, -0.586087235467691, -0.405845151377397,
           -0.207784955007899, 0, 0.207784955007899, 0.405845151377397,
           0.586087235467691, 0.741531185599394, 0.864864423359769,
           0.949107912342758, 0.991455371120813)
  c15 <- c(0.0229353220105292, 0.0630920926299785, 0.10479001032225,
           0.140653259715526, 0.169004726639268, 0.190350578064785,
           0.204432940075299, 0.209482141084728, 0.204432940075299,
           0.190350578064785, 0.169004726639268, 0.140653259715526,
           0.10479001032225, 0.0630920926299785, 0.0229353220105292)
  diff.lim<-ulim - llim
  sum.lim<-ulim + llim
  x15<-t(0.5*(sapply(diff.lim,FUN= function(u) u*t15)+
                sapply(sum.lim,FUN= function(u) u*rep(1,15))))
  y15<-hEPH.assump(x15,kappa=kappa,gam=gam,rho=rho,
                   betat=betat,xx=xx,point=15,prec.bits=prec.bits)
  K15 <- colSums(c15 * y15)
  return(K15*diff.lim/2)
}
##################################################
MKumWPH.recurrent.assump<-function(begin,stop,status,xx,fit.cox,prec.bits=NULL
    ){
  llik.PH.assump0<-function(init,begin,stop,status,xx,xx1,beta,betat,prec.bits
      =NULL){
```

```
  kappa<-exp(init[1])
  gam<-exp(init[2])
  rho<-exp(init[3])
  pred1<-c(as.matrix(xx)%*%beta)
  pred2<-c(as.matrix(xx1)%*%betat)
  m<-sum(status)
  if(!is.null(prec.bits)){
    a0<-mpfr(-(stop^kappa),prec.bits)
  } else{
    a0<--(stop^kappa)
  }
  a<-exp(a0) # a
  a2<-log1p(-a) # log(1-a)
  aa<-log(-expm1(gam*a2)) #log(1-(1-a)^gam)
  log.stop<-log(stop)
  t0<-(kappa-1)*log.stop+(gam-1)*a2+a0-aa
  t1<-status*(t0+pred1+pred2)
  term1<-m*log(kappa)+m*log(gam)+m*log(rho)
  term2<-sum(t1,na.rm=T)
  t2<-gauss_kronrod1(ulim=stop,llim=begin,kappa=kappa,gam=gam,rho=rho,
                     betat=betat,xx=xx,prec.bits=prec.bits)*exp(pred1)
  term3<-sum(t2,na.rm=T)
  lf<-term1+term2-term3
  return(-as.numeric(lf))
}
###########
llik.PH.assump<-function(init,begin,stop,status,xx,xx1,prec.bits=NULL){
  kappa<-exp(init[1])
  gam<-exp(init[2])
  rho<-exp(init[3])
  beta<-init[4:(4+ncol(xx)-1)]
  betat<-init[(4+ncol(xx)):length(init)]
  pred1<-c(as.matrix(xx)%*%beta)
  pred2<-c(as.matrix(xx1)%*%betat)
  m<-sum(status)
  if(!is.null(prec.bits)){
    a0<-mpfr(-(stop^kappa),prec.bits)
  } else{
    a0<--(stop^kappa)
  }
  a<-exp(a0) # a
  a2<-log1p(-a) # log(1-a)
  aa<-log(-expm1(gam*a2)) #log(1-(1-a)^gam)
  log.stop<-log(stop)
  t0<-(kappa-1)*log.stop+(gam-1)*a2+a0-aa
  t1<-status*(t0+pred1+pred2)
  term1<-m*log(kappa)+m*log(gam)+m*log(rho)
  term2<-sum(t1,na.rm=T)
  t2<-gauss_kronrod1(ulim=stop,llim=begin,kappa=kappa,gam=gam,rho=rho,
                     betat=betat,xx=xx,prec.bits=prec.bits)*exp(pred1)
  term3<-sum(t2,na.rm=T)
```

```
  lf<-term1+term2-term3
  return(-as.numeric(lf))
}
###########
xx1<-xx*log(stop)
cox.est<-coef(fit.cox)
beta<-cox.est[1:ncol(xx)]
betat<-cox.est[(ncol(xx)+1):length(cox.est)]
init<-rep(0,3)
options(warn=-1)
fit0<-nlminb(start=init,objective=llik.PH.assump0,
             begin=begin,stop=stop,status=status,xx=xx,xx1=xx1,beta=beta,
               betat=betat,prec.bits=prec.bits)
options(warn=0)
if(fit0$message!="relative convergence (4)"){
  init<-rep(-1,3)
  options(warn=-1)
  fit0<-nlminb(start=init,objective=llik.PH.assump0,
               begin=begin,stop=stop,status=status,xx=xx,xx1=xx1,beta=beta,
                 betat=betat,prec.bits=prec.bits)
  options(warn=0)
}
if(fit0$message!="relative convergence (4)"){
  init<-rep(1,3)
  options(warn=-1)
  fit0<-nlminb(start=init,objective=llik.PH.assump0,
               begin=begin,stop=stop,status=status,xx=xx,xx1=xx1,beta=beta,
                 betat=betat,prec.bits=prec.bits)
  options(warn=0)
}
#######
init<-c(fit0$par,cox.est)
options(warn=-1)
fit1<-nlminb(start=init,objective=llik.PH.assump,
             begin=begin,stop=stop,status=status,xx=xx,xx1=xx1,prec.bits=
               prec.bits)
options(warn=0)
if(fit1$message!="relative convergence (4)"){
  init<-c(rep(-1,3),cox.est)
  options(warn=-1)
  fit1<-nlminb(start=init,objective=llik.PH.assump,
               begin=begin,stop=stop,status=status,xx=xx,xx1=xx1,prec.bits=
                 prec.bits)
  options(warn=0)
}
if(fit1$message!="relative convergence (4)"){
  init<-c(rep(0,3),cox.est)
  options(warn=-1)
  fit1<-nlminb(start=init,objective=llik.PH.assump,
               begin=begin,stop=stop,status=status,xx=xx,xx1=xx1,prec.bits=
                 prec.bits)
```

```
    options(warn=0)
  }
  if(fit1$message!="relative convergence (4)"){
    init<-c(rep(1,3),cox.est)
    options(warn=-1)
    fit1<-nlminb(start=init,objective=llik.PH.assump,
                 begin=begin,stop=stop,status=status,xx=xx,xx1=xx1,prec.bits=
                   prec.bits)
    options(warn=0)
  }
  ##############################
  hess<-fdHess(pars=fit1$par,fun=llik.PH.assump,begin=begin,stop=stop,status=
      status,
               xx=xx,xx1=xx1,prec.bits=prec.bits)$Hessian
  if(is.na(sum(hess))){
    hess<-hessian(func=llik.PH.assump,x=fit1$par,st=st,status=status,xx=xx,xx1
      =xx1)
  }
  if(all(eigen(hess)$values>0) && !is.na(sum(hess))){
    cov.mat<-solve(hess)
  } else{
    warning("MKumWPH covariance matrix is not positive definite")
  }
  ##############
  se<-sqrt(diag(cov.mat))
  z<-fit1$par/se
  p.val<-round(2*pnorm(abs(z),lower.tail=F),digits=4)
  MKumWPH.results0<-cbind(est=fit1$par,se=se,z=z,p=p.val)
  rownames(MKumWPH.results0)[1:3]<-c("logkappa","loggam","logrho")
  MKumWPH.results<-rbind(MKumWPH.results0[4:nrow(MKumWPH.results0),],MKumWPH.
      results0[1:3,])
  ################
  fit.cox0<-coxph(Surv(begin,stop,status)~.,data=xx)
  zp <- cox.zph(fit.cox0, transform= function(time) log(time))
  if(ncol(xx)<=3){
    par(mfrow=c(1,ncol(xx)))
  } else{
    par(mfrow=c(ceiling(ncol(xx)/3),3))
  }
  for(i in 1:ncol(xx)){
    plot(zp[i],xlab="",ylab="")
    abline(MKumWPH.results[c(i,(ncol(xx)+i)),1],col="gray75")
    abline(coef(fit.cox)[c(i,(ncol(xx)+i))],col="gray75",lty=2)
  }
  return(list(MKumWPH.results=MKumWPH.results,convergence=fit1$message,cox.
      results=fit.cox))
}
################################################
# Generate initial values for MKumWPH.
# This is used in the main function i.e. fit.recurrent.model
# begin = start time for each interval of follow-up.
```

```
# stop = stop time for each interval of follow-up.
# status = event status (failure or censored).
# xx = covariate matrix.
################################################
MKumWPH.recurrent.init<-function(begin,stop,status,xx,prec.bits=NULL){
  MKumWPH.recurrent.llik0<-function(init,beta,begin,stop,status,xx,prec.bits=
    NULL){
    kappa<-exp(init[1])
    gam<-exp(init[2])
    rho<-exp(init[3])
    if(!is.null(xx)){
      pred<-as.matrix(xx)%*%beta
    } else{
      pred<-rep(0,length(stop))
    }
    m<-sum(status)
    if(!is.null(prec.bits)){
      a0<-mpfr(-(stop^kappa),prec.bits)
      b0<-mpfr(-(begin^kappa),prec.bits)
    } else{
      a0<--(stop^kappa)
      b0<--(begin^kappa)
    }
    a<-exp(a0) # a
    a2<-log1p(-a) # log(1-a)
    aa<-log(-expm1(gam*a2)) #log(1-(1-a)^gam)
    b<-exp(b0)
    b2<-log1p(-b)
    bb<-log(-expm1(gam*b2))
    t1<-status*((kappa-1)*log(stop)+(gam-1)*a2+a0-aa+pred)
    t2<- -rho*(aa-bb)*exp(pred)
    lf<-m*log(kappa)+m*log(gam)+m*log(rho)+sum(t1,na.rm=T)-sum(t2,na.rm=T)
    return(-as.numeric(lf))
  }
  cox.fit<-coxph(Surv(begin,stop,status)~.,data=xx)
  beta.cox<-coef(cox.fit)
  init1<-rep(0,3)
  init2<-rep(1,3)
  init3<-rep(2,3)
  init4<-rep(-1,3)
  init5<-rep(-2,3)
  conv0<-NULL
  est0<-NULL
  messag<-NULL
  options(warn=-1)
  fit0<-nlminb(start=init1,objective=MKumWPH.recurrent.llik0,beta=beta.cox,
               begin=begin,stop=stop,status=status,xx=xx,prec.bits=prec.bits)
  if(fit0$message=="relative convergence (4)" && is.finite(fit0$objective)){
    conv0<-c(conv0,fit0$objective)
    est0<-rbind(est0,fit0$par)
    messag<-c(messag,fit0$message)
```

```
  }
  fit0<-nlminb(start=init2,objective=MKumWPH.recurrent.llik0,beta=beta.cox,
               begin=begin,stop=stop,status=status,xx=xx,prec.bits=prec.bits)
  if(fit0$message=="relative convergence (4)" && is.finite(fit0$objective)){
    conv0<-c(conv0,fit0$objective)
    est0<-rbind(est0,fit0$par)
    messag<-c(messag,fit0$message)
  }
  fit0<-nlminb(start=init3,objective=MKumWPH.recurrent.llik0,beta=beta.cox,
               begin=begin,stop=stop,status=status,xx=xx,prec.bits=prec.bits)
  if(fit0$message=="relative convergence (4)" && is.finite(fit0$objective)){
    conv0<-c(conv0,fit0$objective)
    est0<-rbind(est0,fit0$par)
    messag<-c(messag,fit0$message)
  }
  fit0<-nlminb(start=init4,objective=MKumWPH.recurrent.llik0,beta=beta.cox,
               begin=begin,stop=stop,status=status,xx=xx,prec.bits=prec.bits)
  if(fit0$message=="relative convergence (4)" && is.finite(fit0$objective)){
    conv0<-c(conv0,fit0$objective)
    est0<-rbind(est0,fit0$par)
    messag<-c(messag,fit0$message)
  }
  fit0<-nlminb(start=init5,objective=MKumWPH.recurrent.llik0,beta=beta.cox,
               begin=begin,stop=stop,status=status,xx=xx,prec.bits=prec.bits)
  if(fit0$message=="relative convergence (4)" && is.finite(fit0$objective)){
    conv0<-c(conv0,fit0$objective)
    est0<-rbind(est0,fit0$par)
    messag<-c(messag,fit0$message)
  }
  options(warn=0)
  if(is.null(messag)){
    return(list(message="non-convergence"))
  }
  conv00<-min(conv0)
  init00<-est0[conv0==conv00,]
  if(length(c(init00))>length(fit0$par)){
    init00<-init00[1,]
  } else{
    init00<-init00
  }
  message<-messag[conv0==conv00]
  if(length(message)>1){
    message<-message[1]
  }
  init0<-c(init00,beta.cox)
  names(init0)<-c("logkappa","loggam","logrho",colnames(xx))
  return(list(init=init0,obj=conv00,message=message))
}
################################################
# Generate initial values for Weibull PH.
# This is used in the main function
```

```
# fit.recurrent.model (see below).
# begin = start time for each interval of follow-up.
# stop = stop time for each interval of follow-up.
# status = event status (failure or censored).
# xx = covariate matrix.
################################################
WPH.recurrent.init<-function(begin,stop,status,xx){
  WPH.recurrent.llik0<-function(init,beta,begin,stop,status,xx){
    kappa<-exp(init[1])
    rho<-exp(init[2])
    if(!is.null(xx)){
      pred<-as.matrix(xx)%*%beta
    } else{
      pred<-rep(0,length(stop))
    }
    m<-sum(status)
    t1<-status*((kappa-1)*log(stop)+pred)
    t2<- rho*(stop^kappa-begin^kappa)*exp(pred)
    lf<-m*log(kappa)+m*log(rho)+sum(t1,na.rm=T)-sum(t2,na.rm=T)
    return(-as.numeric(lf))
  }
  cox.fit<-coxph(Surv(begin,stop,status)~.,data=xx)
  beta.cox<-coef(cox.fit)
  init1<-rep(0,2)
  init2<-rep(1,2)
  init3<-rep(2,2)
  init4<-rep(-1,2)
  init5<-rep(-2,2)
  conv0<-NULL
  est0<-NULL
  messag<-NULL
  options(warn=-1)
  fit0<-nlminb(start=init1,objective=WPH.recurrent.llik0,beta=beta.cox,
               begin=begin,stop=stop,status=status,xx=xx)
  if(fit0$message=="relative convergence (4)" && is.finite(fit0$objective)){
    conv0<-c(conv0,fit0$objective)
    est0<-rbind(est0,fit0$par)
    messag<-c(messag,fit0$message)
  }
  fit0<-nlminb(start=init2,objective=WPH.recurrent.llik0,beta=beta.cox,
               begin=begin,stop=stop,status=status,xx=xx)
  if(fit0$message=="relative convergence (4)" && is.finite(fit0$objective)){
    conv0<-c(conv0,fit0$objective)
    est0<-rbind(est0,fit0$par)
    messag<-c(messag,fit0$message)
  }
  fit0<-nlminb(start=init3,objective=WPH.recurrent.llik0,beta=beta.cox,
               begin=begin,stop=stop,status=status,xx=xx)
  if(fit0$message=="relative convergence (4)" && is.finite(fit0$objective)){
    conv0<-c(conv0,fit0$objective)
    est0<-rbind(est0,fit0$par)
```

```
    messag<-c(messag,fit0$message)
  }
  fit0<-nlminb(start=init4,objective=WPH.recurrent.llik0,beta=beta.cox,
               begin=begin,stop=stop,status=status,xx=xx)
  if(fit0$message=="relative convergence (4)" && is.finite(fit0$objective)){
    conv0<-c(conv0,fit0$objective)
    est0<-rbind(est0,fit0$par)
    messag<-c(messag,fit0$message)
  }
  fit0<-nlminb(start=init5,objective=WPH.recurrent.llik0,beta=beta.cox,
               begin=begin,stop=stop,status=status,xx=xx)
  if(fit0$message=="relative convergence (4)" && is.finite(fit0$objective)){
    conv0<-c(conv0,fit0$objective)
    est0<-rbind(est0,fit0$par)
    messag<-c(messag,fit0$message)
  }
  options(warn=0)
  if(is.null(messag)){
    return(list(message="non-convergence"))
  }
  conv00<-min(conv0)
  init00<-est0[conv0==conv00,]
  if(length(c(init00))>length(fit0$par)){
    init00<-init00[1,]
  } else{
    init00<-init00
  }
  message<-messag[conv0==conv00]
  if(length(message)>1){
    message<-message[1]
  }
  init0<-c(init00,beta.cox)
  names(init0)<-c("logtau","logrho",colnames(xx))
  return(list(init=init0,obj=conv00,message=message))
}

################################################
# Analysis of bladder cancer data
# The data (in counting process format)
# are given in the "survival" package.
################################################
bladder2[1:10,]
id<-bladder2$id
begin<-bladder2$start
stop<-bladder2$stop
status<-bladder2$event
rx<-ifelse(bladder2$rx==1,0,1)
xx<-data.frame(tx=rx,num=bladder2$number,size=bladder2$size)
fit<-fit.recurrent.model(begin=begin,stop=stop,status=status,xx=xx)
fit
```

```
# Residual plots
par(mfrow=c(1,3))
recurrent.diagnostic.plot(fit,begin,stop,status,xx,id,model="MKumWPH",prec.
    bits=NULL)
recurrent.diagnostic.plot(fit,begin,stop,status,xx,id,model="WPH",prec.bits=
    NULL)
recurrent.diagnostic.plot(fit,begin,stop,status,xx,id,model="AG",prec.bits=
    NULL)



# Checking PH assumption
fit.cox<-coxph(Surv(begin,stop,status)~tx+num+size+tt(tx)+tt(num)+tt(size),
               tt = function(x, t, ...) x * log(t),data=xx)

ph.assump<-MKumWPH.recurrent.assump(begin,stop,status,xx,fit.cox,prec.bits=
    NULL)

ph.assump

#########################################################################
```

A.6 Gauss-Legendre Quadrature for Numerical Integration

Definite integrals of a function can be well approximated using a quadrature rule. In general, the approximation is a weighted sum of function values at some specified points, called nodes. Assuming that the range of integration is from -1 to $+1$, a q-point Gaussian quadrature rule uses the approximation

$$\int_{-1}^{1} h(x)dx \approx \sum_{k=1}^{q} w_k h(x_k),$$

where x_k and w_k are nodes and weights, respectively. The quadrature nodes are defined as the roots of a polynomial on $[-1, 1]$ from a class of orthogonal polynomials. For Gauss-Legendre quadrature, Legendre polynomials $P_q(x)$ are used. The weights are then computed using the formula

$$w_i = \frac{2}{(1 - x_k^2)[P'_q(x_k)]^2}.$$

For $q = 5$, the nodes and weights of Gauss-Legendre quadrature are given in Table A.1 (readers may refer to Press et al. (2007) for more details).

Table A.1: Nodes and weights of the 5-point Gauss-Legendre quadrature.

Nodes (x_k)	0.9061798459	−0.9061798459	0.5384693101	−0.5384693101	0
Weights (w_k)	0.2369268851	0.2369268851	0.4786286705	0.4786286705	0.5688888888

To apply a quadrature rule, we must first transform an integral so that its range is from -1 to 1. An integral over $[a, b]$ can be transformed into an integral over $[-1, 1]$ as follows:

$$\int_a^b h(x)dx = \frac{b-a}{2}\int_{-1}^{1} h\Big(\frac{b-a}{2}x + \frac{b+a}{2}\Big)dx.$$

For joint modeling, we use 5-point Gauss-Legendre rule to evaluate the integral in (5.11). Taking $a = 0$ and $b = t$ as in (5.11), the 5-point Gauss-Legendre rule gives

$$\int_0^t h(x)dx = \frac{t}{2}\int_{-1}^{1} h\Big(\frac{t}{2}x + \frac{t}{2}\Big)dx \approx \frac{t}{2}\sum_{k=1}^{5} w_k h\Big(\frac{t}{2}x_k + \frac{t}{2}\Big).$$

In our implementation, we compute $\frac{t_i}{2}x_k + \frac{t_i}{2}$ for each individual i in R. The hazard function is then evaluate at these values in WinBUGS for numerical integration.

A.7 WinBUGS Codes to Fit Joint Models

WinBUGS function to fit joint models using the MKumw distribution is presented below.

```
model
{
    for (i in 1:n) {
        for (j in n1[i]:n2[i]) {
            y[j] ~ dnorm(muy[j], inv.sigSqu)
            muy[j] <- inprod2(alpha[1:p1], xlong[j, 1:p1]) +
                inprod2(u[i, 1:pp1], zlong[j, 1:pp1])
        }
        zeros[i] ~ dpois(zeros.mean[i])
        zeros.mean[i] <- -l[i] + const
        pred1[i] <- inprod2(beta[1:p2], xsurv[i, 1:p2])
        pred2[i] <- phi * inprod2(u[i, 1:pp2], zsurv[i, 1:pp2])
        a0[i] <- 1 - exp(-pow(st[i], kappa))
        a1[i] <- 1/pow(a0[i], (gam - 1))
        a2[i] <- min(0.99999999999999, pow(a0[i], gam))
        a3[i] <- pow(a2[i], (1/gam))
        ind[i] <- equals(a1[i], 1)
        term1[i] <- status[i] * (log(kappa) + log(rho) + log(gam) +
            (kappa - 1) * log(st[i]) - pow(st[i], kappa) + ind[i] *
            pow(st[i], kappa) + (1 - ind[i]) * ((gam - 1) * log(a3[i])
                -
            log(1 - a2[i])) + pred1[i] + pred2[i])
        for (k in 1:quad.points) {
            b0[i, k] <- 1 - exp(-pow(x15[i, k], kappa))
```

```
            b1[i, k] <- 1/pow(b0[i, k], (gam - 1))
            b2[i, k] <- min(0.99999999999999, pow(b0[i, k], gam))
            b3[i, k] <- pow(b2[i, k], (1/gam))
            ind1[i, k] <- equals(b1[i, k], 1)
            kk15[i, k] <- c15[k] * exp((kappa - 1) * log(x15[i,
                k]) - pow(x15[i, k], kappa) + ind1[i, k] * pow(x15[i,
                k], kappa) + (1 - ind1[i, k]) * ((gam - 1) *
                log(b3[i, k]) - log(1 - b2[i, k])) + phi * inprod2(u[i,
                1:pp2], xx15[i, 1:pp2, k]))
        }
        term2[i] <- kappa * gam * rho * sum(kk15[i, ]) * st[i] *
            exp(pred1[i])/2
        l[i] <- term1[i] - term2[i]
        u[i, 1:pp1] ~ dmnorm(U0[], inv.Sigma[, ])
    }
    inv.Sigma[1:pp1, 1:pp1] ~ dwish(R[, ], w.df)
    alpha[1:p1] ~ dmnorm(alpha.mu[], iSigma1[, ])
    beta[1:p2] ~ dmnorm(beta.mu[], iSigma2[, ])
    phi ~ dnorm(prior.phi.mu, prior.phi.tau)
    inv.sigSqu ~ dgamma(prior.tauz1, prior.tauz2)
    kappa1 ~ dgamma(prior.kappa1, prior.kappa2)
    gam1 ~ dgamma(prior.gam1, prior.gam2)
    rho ~ dgamma(prior.rho1, prior.rho2)
    kappa <- 1/kappa1
    gam <- 1/gam1
    logkappa <- log(kappa)
    loggam <- log(gam)
    logrho <- log(rho)
}
```

WinBUGS function to fit joint models using the Weibull distribution is presented below.

```
model
{
    for (i in 1:n) {
        for (j in n1[i]:n2[i]) {
            y[j] ~ dnorm(muy[j], inv.sigSqu)
            muy[j] <- inprod2(alpha[1:p1], xlong[j, 1:p1]) +
                inprod2(u[i, 1:pp1], zlong[j, 1:pp1])
        }
        zeros[i] ~ dpois(zeros.mean[i])
        zeros.mean[i] <- -l[i] + const
        pred1[i] <- inprod2(beta[1:p2], xsurv[i, 1:p2])
        pred2[i] <- phi * inprod2(u[i, 1:pp2], zsurv[i, 1:pp2])
```

```
        term1[i] <- status[i] * (log(kappa) + log(rho) + (kappa -
            1) * log(st[i]) + pred1[i] + pred2[i])
        for (k in 1:quad.points) {
            kk15[i, k] <- c15[k] * pow(x15[i, k], (kappa - 1)) *
                exp(phi * inprod2(u[i, 1:pp2], xx15[i, 1:pp2,
                  k]))
        }
        term2[i] <- kappa * rho * sum(kk15[i, ]) * st[i] * exp(pred1[i
           ])/2
        l[i] <- term1[i] - term2[i]
        u[i, 1:pp1] ~ dmnorm(U0[], inv.Sigma[, ])
    }
    inv.Sigma[1:pp1, 1:pp1] ~ dwish(R[, ], w.df)
    alpha[1:p1] ~ dmnorm(alpha.mu[], iSigma1[, ])
    beta[1:p2] ~ dmnorm(beta.mu[], iSigma2[, ])
    phi ~ dnorm(prior.phi.mu, prior.phi.tau)
    inv.sigSqu ~ dgamma(prior.tauz1, prior.tauz2)
    kappa1 ~ dgamma(prior.kappa1, prior.kappa2)
    rho ~ dgamma(prior.rho1, prior.rho2)
    kappa <- 1/kappa1
    logkappa <- log(kappa)
    logrho <- log(rho)
}
```

REFERENCES

Aalen, O. (1978). Nonparametric inference for a family of counting processes. *The Annals of Statistics*, 6(4):701–726. 12

Abbott, P. (2005). Tricks of the trade: Legendre-gauss quadrature. *Mathematica Journal*, 9:689–691. 97

Abrams, D., Goldman, A., Launer, C., Korvick, J., Neaton, J., Crane, L., Grodesky, M., Wakefield, S., Muth, K., Kornegay, S., Cohn, D., Harris, A., Luskin-Hawk, R., Markowitz, N., Sampson, J., Thompson, M., and Deyton, L. (1994). Comparative trial of didanosine and zalcitabine in patients with human immunodeficiency virus infection who are intolerant of or have failed zidovudine therapy. *New England Journal of Medicine*, 330:657–662. 103

Akaike, H. (1974). A new look at the statistical model identification. *Automatic Control, IEEE Transactions on*, 19(6):716–723. 30, 31, 47, 76

Andersen, P. K., Borgan, O., Gill, R. D., and Keiding, N. (2012). *Statistical models based on counting processes*. Springer Science & Business Media. 32

Andersen, P. K. and Gill, R. D. (1982). Cox's regression model for counting processes: a large sample study. *The annals of statistics*, pages 1100–1120. 32, 33, 86, 90

Barlow, R. E. and Campo, R. A. (1975). Total time on test processes and applications to failure data analysis. Technical report, CALIFORNIA UNIV BERKELEY OPERATIONS RESEARCH CENTER. 48

Bennett, S. (1983). Log-logistic regression models for survival data. *Applied Statistics*, pages 165–171. 15

Brooks, S. P. and Gelman, A. (1998). General methods for monitoring convergence of iterative simulations. *Journal of computational and graphical statistics*, 7(4):434–455. 38

Burnham, K. P. and Anderson, D. R. (2002). *Model selection and multimodel inference: a practical information-theoretic approach*. Springer Science & Business Media. 31

Burnham, K. P. and Anderson, D. R. (2003). *Model selection and multimodel inference: a practical information-theoretic approach*. Springer Science & Business Media. 47

Byar, D. (1980). The veterans administration study of chemoprophylaxis for recurrent stage i bladder tumours: comparisons of placebo, pyridoxine and topical thiotepa. In *Bladder tumors and other topics in urological oncology*, pages 363–370. Springer. 92

Chaubey, Y. P. and Zhang, R. (2015). An extension of chen's family of survival distributions with bathtub shape or increasing hazard rate function. *Communications in Statistics-Theory and Methods*, 44(19):4049–4064. 18

Chen, Z. (2000). A new two-parameter lifetime distribution with bathtub shape or increasing failure rate function. *Statistics & Probability Letters*, 49(2):155–161. 18

Collett, D. (2003). *Modelling survival data in medical research*. Chapman & Hall/CRC. 3, 8, 10, 23, 28, 29, 30, 41

Congdon, P. (2014). *Applied bayesian modelling*. John Wiley & Sons. 70

Cook, R. J. and Lawless, J. (2007). *The statistical analysis of recurrent events*. Springer Science & Business Media. 26, 32, 33, 63, 84, 86, 87, 89, 90, 91, 109, 116

Cordeiro, G. M. and de Castro, M. (2011). A new family of generalized distributions. *Journal of statistical computation and simulation*, 81(7):883–898. 66

Cordeiro, G. M., Ortega, E. M., and Nadarajah, S. (2010). The kumaraswamy weibull distribution with application to failure data. *Journal of the Franklin Institute*, 347(8):1399–1429. 6, 7, 63, 66, 113

Cox, D. R. (1972). Regression models and life-tables. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 187–220. 3, 22, 23, 24, 41, 64

Davis, P. J. and Rabinowitz, P. (2007). *Methods of numerical integration*. Courier Corporation. 76

Efron, B. (1977). The efficiency of cox's likelihood function for censored data. *Journal of the American statistical Association*, 72(359):557–565. 3, 41, 83, 111

Efron, B. (1988). Logistic regression, survival analysis, and the kaplan-meier curve. *Journal of the American Statistical Association*, 83(402):414–425. 49, 64

Fitzmaurice, G. M., Laird, N. M., and Ware, J. H. (2011). *Applied Longitudinal Analysis*, volume 745. John Wiley & Sons. 35, 98

Fuchs, H. J., Borowitz, D. S., Christiansen, D. H., Morris, E. M., Nash, M. L., Ramsey, B. W., Rosenstein, B. J., Smith, A. L., and Wohl, M. E. (1994). Effect of aerosolized recombinant human dnase on exacerbations of respiratory symptoms and on pulmonary function in patients with cystic fibrosis. *New England Journal of Medicine*, 331(10):637–642. 77

Gelman, A. (1995). Inference and monitoring convergence. In Gilks, W., Richardson, S., and Spiegelhalter, D., editors, *Markov Chain Monte Carlo in Practice*, pages 131–143. Chapman and Hall, London. 100

Gelman, A. and Rubin, D. (1992). Inference from iterative simulation using multiple sequences. *Statistical Science*, 7:457–472. 38, 100

Gelman, A., Stern, H. S., Carlin, J. B., Dunson, D. B., Vehtari, A., and Rubin, D. B. (2013). *Bayesian data analysis*. Chapman and Hall/CRC. 36, 37, 98

Gilbert, P. and Varadhan, R. (2016). numderiv: Accurate numerical derivatives [software]. 72, 73

Givens, G. and Hoeting, J. (2005). *Computational Statistics*. John Wiley & Sons, New Jersey. 99

Grambsch, P. M. and Therneau, T. M. (1994). Proportional hazards tests and diagnostics based on weighted residuals. *Biometrika*, 81(3):515–526. 29, 30, 75

Guo, X. and Carlin, B. (2004). Separate and joint modeling of longitudinal and event time data using standard computer packages. *The American Statistician*, 28:1–9. 35, 88, 96

Henderson, R., Diggle, P., , and Dobson, A. (2000). Joint modelling of longitudianl measurements and event time data. *Biostatistics*, 1:465–480. 35, 88, 95

Hjort, N. L. (1992). On inference in parametric survival data models. *International Statistical Review / Revue Internationale de Statistique*, 60(3):355–387. 4, 40, 64

Hsieh, F., Tseng, Y.-K., and Wang, J.-L. (2006). Joint modeling of survival and longitudinal data: likelihood approach revisited. *Biometrics*, 62(4):1037–1043. 3, 41, 65, 80, 83, 88, 111

Hwang, B. S. and Pennell, M. L. (2014). Semiparametric bayesian joint modeling of a binary and continuous outcome with applications in toxicological risk assessment. *Statistics in medicine*, 33(7):1162–1175. 3, 65

Jackson, C. H. (2016). flexsurv: a platform for parametric survival modeling in r. *Journal of Statistical Software*, 70. 17

Kalbfleisch, J. (2002). The statistical analysis of failure time data. 41, 42

Kalbfleisch, J. D. and Prentice, R. L. (2002). *The statistical analysis of failure time data*. John Wiley & Sons. 2, 4, 19, 23, 63

Kaplan, E. L. and Meier, P. (1958). Nonparametric estimation from incomplete observations. *Journal of the American statistical association*, 53(282):457–481. 11, 47, 74

Khan, S. A. (2017). Exponentiated weibull regression for time-to-event data. *Lifetime data analysis*, 24(2):328–354. 18, 77, 78, 81

Khan, S. A. and Khosa, S. K. (2016). Generalized log-logistic proportional hazard model with applications in survival analysis. *Journal of Statistical Distributions and Applications*, 3(1):16. 40, 65, 113, 119

Kleiber, C. and Kotz, S. (2003). *Statistical size distributions in economics and actuarial sciences*. John Wiley & Sons. 17

Klein, J. P. and Moeschberger, M. L. (2003). *Survival analysis: techniques for censored and truncated data*. Springer Science & Business Media. 1, 8, 50

Klein, J. P. and Moeschberger, M. L. (2005). *Survival analysis: techniques for censored and truncated data*. Springer Science & Business Media. 3

Kleinbaum, D. G. and Klein, M. (2012). *Survival analysis: A self-learning text*. Springer-Verlag, New York. 18, 27, 28, 63, 64, 75, 76

Laird, N. M. and Ware, J. H. (1982). Random-effects models for longitudinal data. *Biometrics*, pages 963–974. 34

Langberg, N. A., Leon, R. V., Proschan, F., et al. (1980). Characterization of nonparametric classes of life distributions. *The Annals of Probability*, 8(6):1163–1170. 48

Langlands, A. O., Pocock, S. J., Kerr, G. R., and Gore, S. M. (1979). Long-term survival of patients with breast cancer: a study of the curability of the disease. *Br med J*, 2(6200):1247–1251. 15

Lawless, J. F. (2003). *Statistical models and methods for lifetime data*. John Wiley & Sons. 3, 4, 8, 10, 11, 15, 19, 21, 25, 28, 29, 41, 42, 47, 48, 52, 60, 64, 65, 74, 75, 77, 112

Lee, E. T. and Wang, J. (2003). *Statistical methods for survival data analysis*. John Wiley & Sons. 1

Lehmann, E. L. (1953). The power of rank tests. *The Annals of Mathematical Statistics*, 24(1):23–43. 3

Liang, K.-Y. and Zeger, S. L. (1986). Longitudinal data analysis using generalized linear models. *Biometrika*, 73(1):13–22. 34

Lin, D. (1994). Cox regression analysis of multivariate failure time data: the marginal approach. *Statistics in medicine*, 13(21):2233–2247. 86

Lunn, D., Thomas, A., Best, N., and Spiegelhalter, D. (2000). Winbugs – a bayesian modelling framework: Concepts, structure, and extensibility. *Statistics and Computing*, 10:325–337. 36, 89, 99, 101

Maechler, M. (2016). Rmpfr: R mpfr-multiple precision floating-point reliable. r package version 0.6-1. 73, 115

Mäechler, M. (2019). *Rmpfr: R MPFR – Multiple Precision Floating-Point Reliable*. R package version 0.7-2. 102

Mudholkar, G. S. and Srivastava, D. K. (1993). Exponentiated weibull family for analyzing bathtub failure-rate data. *IEEE Transactions on Reliability*, 42(2):299–302. 17, 18

Mudholkar, G. S., Srivastava, D. K., and Kollia, G. D. (1996). A generalization of the weibull distribution with application to the analysis of survival data. *Journal of the American Statistical Association*, 91(436):1575–1583. 4, 18, 42, 48, 65, 68, 81

Nardi, A. and Schemper, M. (2003). Comparing cox and parametric models in clinical studies. *Statistics in Medicine*, 22(23):3597–3610. 4, 41

Nelson, W. (1972). Theory and applications of hazard plotting for censored failure data. *Technometrics*, 14(4):945–966. 11, 12

Oakes, D. (1977). The asymptotic information in censored survival data. *Biometrika*, 64(3):441–448. 3, 41, 64, 83, 111

Pham, T. and Almhana, J. (1995). The generalized gamma distribution: its hazard rate and stress-strength model. *IEEE transactions on reliability*, 44(3):392–397. 17

Pike, M. (1966). A method of analysis of a certain class of experiments in carcinogenesis. *Biometrics*, 22(1):142–161. 52

Plummer, M., Best, N., Cowles, K., and Vines, K. (2006). Coda: Convergence diagnosis and output analysis for mcmc. *R News*, 6(1):7–11. 102

Press, W., Teukolsky, S., Vetterling, W., and Flannery, B. (2007). *Numerical Recipes: The Art of Scientific Computing*. Cambridge University Press, New York, 3rd edition. 35, 88, 91, 157

R Core Team (2016). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. 30, 45, 67, 72

R Core Team (2018). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. 89, 101

Reed, W. J. (2011). A flexible parametric survival model which allows a bathtub-shaped hazard rate function. *Journal of Applied Statistics*, 38(8):1665–1680. 65

Reid, N. (1994). A conversation with sir david cox. *Statistical Science*, 9(3):439–455. 4, 65

Riggelsen, C. (2008). *Approximation methods for efficient learning of Bayesian networks*, volume 168. IOS Press. 37

Rizopoulos, D. (2012). *Joint models for longitudinal and time-to-event data: With applications in R*. CRC Press. 3, 35, 41, 46, 54, 55, 64, 65, 84, 86, 88, 95, 101, 109, 110, 111

Rizopoulos, D. et al. (2010). Jm: An r package for the joint modelling of longitudinal and time-to-event data. *Journal of Statistical Software*, 35(9):1–33. 46, 118

Royston, P. and Parmar, M. K. (2002). Flexible parametric proportional-hazards and proportional-odds models for censored survival data, with application to prognostic modelling and estimation of treatment effects. *Statistics in medicine*, 21(15):2175–2197. 64, 65, 111

Schoenfeld, D. (1982). Partial residuals for the proportional hazards regression model. *Biometrika*, 69(1):239–241. 27, 29

Schwarz, G. et al. (1978). Estimating the dimension of a model. *The annals of statistics*, 6(2):461–464. 30, 38

Self, S. and Pawitan, Y. (1992). Modeling a marker of disease progression and onset of disease. In *AIDS Epidemiology*, pages 231–255. Springer. 35

Spiegelhalter, D., Thomas, A., Best, N., and Lunn, D. (2003). *WinBUGS User Manual*. MRC Biostatistics Unit, University of Cambridge. 98, 99

Spiegelhalter, D. J., Best, N. G., Carlin, B. P., and Linde, A. (2014). The deviance information criterion: 12 years on. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 76(3):485–493. 39

Spiegelhalter, D. J., Best, N. G., Carlin, B. P., and Van Der Linde, A. (2002). Bayesian measures of model complexity and fit. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 64(4):583–639. 30, 38, 100, 101

Stacy, E. W. (1962). A generalization of the gamma distribution. *The Annals of mathematical statistics*, pages 1187–1192. 17

Sturtz, S., Ligges, W., and Gelman, A. (2005). R2winbugs: A package for running winbugs from r. *Journal of Statistical Software*, 12(3):1–16. 89, 102

Therneau, T. M. (2015). A package for survival analysis in s. 2015. version 2.38. 92

Therneau, T. M. and Grambsch, P. M. (2013). *Modeling survival data: extending the Cox model*. Springer Science & Business Media. 32, 34

Therneau, T. M., Grambsch, P. M., and Fleming, T. R. (1990). Martingale-based residuals for survival models. *Biometrika*, 77(1):147–160. 33

Thom, H. C. (1958). A note on the gamma distribution. *Monthly Weather Review*, 86(4):117–122. 16

Tsiatis, A., Degruttola, V., and Wulfsohn, M. (1995). Modeling the relationship of survival to longitudinal data measured with error. applications to survival and cd4 counts in patients with aids. *Journal of the American Statistical Association*, 90(429):27–37. 35

Tsiatis, A. A. and Davidian, M. (2004). Joint modeling of longitudinal and time-to-event data: an overview. *Statistica Sinica*, 14(3):809–834. 35, 86, 87

Wang, Y., Hossain, A. M., and Zimmer, W. J. (2008). Useful properties of the three-parameter burr xii distribution. *Applied Statistics Research Progress, Editor: Mohammad Ahsanullah*, pages 11–20. 42

Wang, Y. and Taylor, J. M. G. (2001). Jointly modeling longitudinal and event time data with application to acquired immunodeficiency syndrome. *Journal of the American Statistical Association*, 96(455):895–905. 35

Watanabe, S. (2010). Asymptotic equivalence of bayes cross validation and widely applicable information criterion in singular learning theory. *Journal of Machine Learning Research*, 11(Dec):3571–3594. 38

Wei, L. (1992). The accelerated failure time model: a useful alternative to the cox regression model in survival analysis. *Statistics in medicine*, 11(14-15):1871–1879. 2

Wei, L.-J., Lin, D. Y., and Weissfeld, L. (1989). Regression analysis of multivariate incomplete failure time data by modeling marginal distributions. *Journal of the American statistical association*, 84(408):1065–1073. 86

Weibull, W. (1951). Wide applicability. *Journal of applied mechanics*, 103:33. 13

Williamson, P., Kolamunnage-Dona, R., Philipson, P., and Marson, A. (2008). Joint modelling of longitudinal and competing risks data. *Statistics in medicine*, 27(30):6426–6438. 88

Wu, L., Liu, W., Yi, G. Y., and Huang, Y. (2012). Analysis of longitudinal and survival data: joint modeling, inference methods, and issues. *Journal of Probability and Statistics*, 2012. 88, 95

Wulfsohn, M. S. and Tsiatis, A. A. (1997). A joint model for survival and longitudinal data measured with error. *Biometrics*, pages 330–339. 3, 35, 41, 46, 64

Xu, J. and Zeger, S. L. (2001). Joint analysis of longitudinal data comprising repeated measures and times to events. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 50(3):375–387. 35